

# Simulation der Phakoemulsifikation im Augenoperationssimulator Eyesi

Inauguraldissertation  
zur Erlangung des akademischen Grades  
eines Doktors der Naturwissenschaften  
der Universität Mannheim

vorgelegt von

Dipl.-Inf.  
Ole Jan Jakubik  
aus Hockenheim

Mannheim, 2009

Dekan: Professor Dr. Felix Freiling, Universität Mannheim  
Referent: Professor Dr. Reinhard Männer, Universität Heidelberg  
Korreferent: Professor Dr. med. Frank Koch, Goethe-Universität Frankfurt am Main

Tag der mündlichen Prüfung: 23. März 2010

*für meine Familie*





„The author will also find a way to remove cataract through a tiny incision, eliminating the need for hospitalization, general anesthesia, and dramatically shortening the recovery period.“

*Charles D. Kelman, Erfinder der Phakoemulsifikation*



# Danksagung

Allen zu danken, die einen Beitrag dazu geleistet haben, dass es zu dieser Arbeit kam, ist leider nicht möglich. Jedoch gab es in den letzten Jahren einige Personen, denen besonderer Dank gilt.

Mein besonderer Dank gilt Herrn Prof. Dr. Reinhard Männer, der mir diese Promotion ermöglicht hat. Ebenso möchte ich Herrn Prof. Dr. med. Frank Koch danken, der sich als Korreferent zur Verfügung gestellt hat und mich beim Buchbeitrag „Modellbasierte Methoden zur Ausbildung von Chirurgen in der Ophthalmologie“ mit seinem Fachwissen unterstützte.

Dank gilt auch meinen Kollegen der ViPA Arbeitsgruppe des LS Informatik V/ICM, mit denen ich jederzeit kompetente Fachgespräche führen konnte. Meinem ehemaligen Büromitbenutzer Klaus Rieger möchte ich für viele mehr oder weniger fachliche Diskussionen zu allen Themen danken, denn nur wenn man ab und zu in die Ferne schweift, behält man den Blick für das Wesentliche. Unseren beiden Sekretärinnen Christiane Glasbrenner und Andrea Seeger gilt mein Dank für ihre Unterstützung bei jeglichem Umgang mit der Verwaltung.

Bedanken möchte ich mich bei der VRmagic GmbH, die durch die Kooperation mit der Universität Mannheim diese Arbeit ermöglicht hat. Dieser Dank schließt die mir zugestandene Freiheit und Unterstützung bei der Entwicklung der Operationsmaschinen-Simulation und der Trainingsmodule ein.

Ein besonderer Dank gilt Dr. Johannes Grimm, der mich in die Kunst der Simulation eingeführt hat und Dr. Clemens Wagner, von dem ich viel über Software-Entwicklung gelernt habe. Ebenso möchte ich mich bei Joachim Schlipper für die vielen Diskussionen, Anregungen und die Hilfe bei der Integration der Operationsmaschinen-Simulation in die Eyesi-Software bedanken.

Ich danke Herrn Montag von der Geuder AG, der mir viele Fragen rund um die Operationsmaschinen beantworten konnte.

Außerdem danke ich allen die meine Arbeit Korrektur gelesen haben, vor allem Christian Ripka, der sich als Außenstehender dazu bereit erklärt hatte.

Marlene, danke für alles.



# Zusammenfassung

Diese Arbeit beschäftigt sich mit der Simulation der Phakoemulsifikation zur Ausbildung von Augenchirurgen. Besonderer Wert wurde auf die Übertragung der für die reale Operation benötigten Phakomaschine in die virtuelle Realität gelegt. Die Phakomaschine wird zum Verflüssigen und Entfernen der Linse aus dem Auge verwendet. Für eine erfolgreiche Operation muss sich der Chirurg bestens mit der Steuerung der Phakomaschine auskennen. Die Erweiterung eines Simulators um solch eine Phakomaschinen-Simulation ermöglicht es dem Lernenden, unter realistischen Bedingungen, wie sie auch im Operationssaal vorkommen, zu lernen.

Bisherige Veröffentlichungen vernachlässigen die Übertragung der Phakomaschine in die virtuelle Realität. Diese Arbeit behebt diesen Missstand. Es wurde ein allgemeines physikalisches Modell der Phakomaschine entwickelt. Das Modell berücksichtigt alle relevanten Maschineneigenschaften wie Irrigation, Aspiration, verschiedene Pumpentypen, Schläuche, Ultraschall und Schnittrate bei der Vitrektomie.

In der Realität hat die Verwendung der Phakomaschine Auswirkungen auf die Stabilität der Vorderkammer. Dies wird in anderen Veröffentlichungen nicht näher untersucht. Hier setzt das vorgestellte flussbasierte Druck- und Volumenmodell für die virtuelle Augumgebung an. Es berücksichtigt das Druck- und Volumenverhalten über die Compliance der Vorderkammer.

Es wurden Modelle für die Interaktionen zwischen den Instrumenten und dem simulierten Gewebe entwickelt. Dies sind neben mechanischen und strömungsabhängigen Interaktionen auch Modelle zur Gewebedestruktion. Keine der aktuellen Veröffentlichungen geht genauer auf die benutzten Modelle zur Gewebedestruktion ein. Hier wird ein vakuum- und ultraschallabhängiges Modell vorgeschlagen, das beide Auswirkungen abhängig von den Maschineneinstellungen verarbeitet.

Die vorgestellten Methoden wurden als Erweiterungen für den Augenoperationssimulator Eyesi implementiert. Somit ist Eyesi der erste Augenoperationssimulator mit einer Operationsmaschinen-Simulation und deren Auswirkungen auf die virtuelle Augumgebung.

Zur Erkenntnisgewinnung wurde zunächst ein Prototyp der Phakoemulsifikation implementiert. Als konkrete Anwendung wurden zwei Trainingsmodule entwickelt. Das erste ist ein abstraktes Trainingsmodul, das dem Lernenden den Umgang mit der Maschine näher bringt. Das zweite Trainingsmodul ist die Simulation der Phakoemulsifikation.



# Inhaltsverzeichnis

<b>1</b>	<b>Einführung</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Computergestützte medizinische Simulatoren . . . . .	2
1.3	Fragestellung . . . . .	4
1.4	Überblick über die Arbeit . . . . .	4
<b>2</b>	<b>Medizinischer Hintergrund</b>	<b>7</b>
2.1	Der Augapfel . . . . .	7
2.2	Die Katarakt . . . . .	8
2.3	Die Kataraktoperation . . . . .	9
2.3.1	Phakoemulsifikation . . . . .	11
2.3.2	Das Instrumentarium . . . . .	12
2.3.3	Anforderungen an die Ausbildung . . . . .	14
<b>3</b>	<b>Grundlagen der Simulation</b>	<b>15</b>
3.1	Szene . . . . .	16
3.2	Modellbildung . . . . .	17
3.2.1	Festkörpersimulation . . . . .	18
3.2.2	Simulation deformierbarer Objekte . . . . .	19
3.2.3	Flüssigkeitssimulation . . . . .	24
3.3	Numerische Integration . . . . .	27
3.4	Kollisionsbehandlung . . . . .	29
3.4.1	Kollisionserkennung . . . . .	29
3.4.2	Kollisionsantwort . . . . .	36
3.5	Modellierung topologischer Veränderungen . . . . .	41
<b>4</b>	<b>Augenchirurgische Simulatoren</b>	<b>43</b>
4.1	PhacoVision . . . . .	43
4.2	A Cataract Tele-Surgery . . . . .	44
4.3	EYESIM: Cataract Surgery Simulator . . . . .	45

4.4	SOFA::EYE . . . . .	46
4.5	Eyesi . . . . .	47
4.5.1	Hardware . . . . .	49
4.5.2	Software . . . . .	51
4.5.3	Anerkennung in der Ausbildung . . . . .	52
4.6	Abgrenzung der Arbeit . . . . .	53
<b>5</b>	<b>Die Operationsmaschine</b>	<b>55</b>
5.1	Physik der Phakoemulsifikation . . . . .	56
5.2	Allgemeiner Aufbau der Operationsmaschine . . . . .	57
5.3	Die Komponenten . . . . .	58
5.4	Die Maschinen-Simulation . . . . .	69
5.4.1	Fußpedal . . . . .	71
5.4.2	Irrigation . . . . .	71
5.4.3	Aspiration . . . . .	73
5.4.4	Instrumentenspezifische Eigenschaften . . . . .	83
5.4.5	Ablauf eines Simulationsschrittes . . . . .	87
5.5	Ergebnisse . . . . .	89
5.5.1	Irrigation . . . . .	90
5.5.2	Aspiration . . . . .	91
5.5.3	Ultraschallmodulation . . . . .	98
5.5.4	Vitrektomie . . . . .	101
5.6	Diskussion . . . . .	102
<b>6</b>	<b>Gewebeinteraktion mit dem Instrumentarium</b>	<b>105</b>
6.1	Mechanische Interaktion . . . . .	107
6.2	Irrigation und Aspiration . . . . .	109
6.2.1	Druck- und Volumenmodell der Vorderkammer . . . . .	109
6.2.2	Das Flussfeld - Auswirkungen auf die Umgebung . . . . .	114
6.3	Verstopfen der Instrumentspitze . . . . .	121
6.4	Ablösen des Gewebes . . . . .	122
6.4.1	Ultraschall . . . . .	122
6.4.2	Vakuum . . . . .	123
6.5	Absaugen des Gewebes . . . . .	124
6.6	Erweiterungen für den hinteren Augenabschnitt . . . . .	124
6.7	Ergebnisse . . . . .	125
6.7.1	Vorderkammer . . . . .	125



6.7.2	Hinterer Augenabschnitt . . . . .	132
6.7.3	Parameter für Eyesi . . . . .	134
6.8	Diskussion . . . . .	134
<b>7</b>	<b>Software-Architektur</b>	<b>137</b>
7.1	Stand der Technik . . . . .	137
7.2	Software-Erweiterungen . . . . .	139
7.2.1	Maschinen-Simulation - ORMachine . . . . .	143
7.2.2	Eyesi . . . . .	144
7.2.3	Maschinen-GUI . . . . .	145
7.3	Maschinentester . . . . .	147
7.4	Diskussion . . . . .	150
<b>8</b>	<b>Trainingsmodule</b>	<b>151</b>
8.1	Phako - Der erste Prototyp . . . . .	151
8.1.1	Anforderungen . . . . .	152
8.1.2	Umsetzung . . . . .	152
8.1.3	Ergebnisse . . . . .	152
8.2	Phako-Training . . . . .	154
8.2.1	Anforderungen . . . . .	154
8.2.2	Umsetzung . . . . .	155
8.2.3	Ergebnisse . . . . .	157
8.3	Phako - Divide and Conquer . . . . .	159
8.3.1	Anforderungen . . . . .	159
8.3.2	Umsetzung . . . . .	160
8.3.3	Ergebnisse . . . . .	161
8.4	Vitreoretinale Trainingsmodule . . . . .	163
8.5	Diskussion . . . . .	163
<b>9</b>	<b>Diskussion und Ausblick</b>	<b>167</b>
	<b>Abbildungsverzeichnis</b>	<b>171</b>
	<b>Tabellenverzeichnis</b>	<b>173</b>
	<b>Literaturverzeichnis</b>	<b>175</b>
	<b>Abkürzungsverzeichnis</b>	<b>185</b>



# Einführung

## 1.1 Motivation

In der heutigen Medizin stellen minimal-invasive Eingriffe keine Seltenheit mehr dar. Diese Art der Operation ist für den Patienten schonender, da die entstandenen Wunden wesentlich kleiner als bei der offenen Chirurgie sind. Für den Operateur sind sie jedoch schwieriger zu erlernen. Er kann das Operationsgebiet über die Instrumente nur indirekt fühlen. Ebenso vermitteln bildgebende Systeme meist nur einen zweidimensionalen visuellen Eindruck. Die heutige Art der Kataraktoperation gehört zu der Kategorie der minimal-invasiven Eingriffe. Der Zweck der Operation ist, die getrübte Linse aus dem Auge zu entfernen und durch eine Kunstlinse zu ersetzen. Ein operativer Eingriff ist die einzige Möglichkeit der Therapie. Allein in Deutschland werden jedes Jahr ca. 400.000 Operationen durchgeführt [101] und [103]. Um diesen Eingriff zu erlernen, ist jahrelanges Training von Nöten. In der klassischen Aus- und Fortbildung werden hauptsächlich Schweineaugen benutzt. Diese müssen für das Training speziell vorbereitet werden, was eine lange Präparationszeit erfordert. Das Verhalten des toten Gewebes entspricht nicht dem des lebenden. Die Anatomie dieser Augen weicht von der des menschlichen ab. Meistens weisen Schweineaugen keinen Katarakt auf, da die Tiere zu jung geschlachtet werden.

Doch welche Alternativen gibt es für die Ausbildung? Eine viel versprechende Möglichkeit ist das Training mittels computergestützter Simulation. Mit immer fortschreitender Computertechnologie ergab sich die Möglichkeit, das Verhalten der Realität in Echtzeit nachzubilden. In der so genannten virtuellen Realität können alle oben genannten Schwierigkeiten umgangen werden. Jedoch müssen andere gelöst werden.

## 1.2 Computergestützte medizinische Simulatoren

Die vorliegende Arbeit beschäftigt sich mit der Erweiterung eines computergestützten Operationssimulators zur Aus- und Fortbildung von Augenchirurgen. Um die Relevanz von Trainingssimulatoren in der medizinischen Ausbildung zu verdeutlichen, sollen diese hier kurz besprochen werden. Im Allgemeinen kann ein Simulator als ein Modell aufgefasst werden, das auf eine bestimmte Handlung mit einer plausiblen Antwort reagiert.

Das wohl bekannteste Beispiel sind Flugsimulatoren, die in der Pilotenausbildung eingesetzt werden. Dabei reagiert eine mechanisch gelagerte Flugzeugkabel auf die Eingaben des Piloten. Durch eine vollständige Abschottung von der Umgebung und der Präsentation einer virtuellen Umgebung stellt sich beim Piloten das Gefühl des realen Fliegens ein. Solange die Reaktion des Simulators plausibel ist, wird dieser Eindruck aufrechterhalten.

Ein häufig verwendeter Begriff im Umfeld der computergestützten Simulatoren ist der der *virtuellen Realität* (kurz VR). Burdea definiert in [15] die virtuelle Realität folgendermaßen:

„Virtual Reality is a high-end user-computer interface that involves real-time simulation and interactions through multiple sensorial channels. These sensorial modalities are visual, auditory, tactile, smell, and taste.“

Aus dieser Definition leitet er die drei „I“ der virtuellen Realität ab. Diese sind die *Interaktion*, die *Immersion* und die *Imagination*. Die ersten beiden gehen direkt aus der Definition hervor, da ein Interface benötigt wird, um die Benutzerinteraktionen aufzunehmen. Die Immersion, das Eintauchen, wird über die Ausgabekanäle erreicht. Jedoch erst durch die Imagination setzt sich alles im Kopf des Benutzers zusammen.

Damit die drei „I“ erreicht werden können, müssen neben einer ausreichend genauen computergrafischen Darstellung, Echtzeitfähigkeit, geringe Latenz, sowie plausible physikalische Reaktionen gewahrt bleiben. Einen guten Überblick über dieses Thema bietet die Arbeit von Wagner [99].

Echtzeitfähigkeit bedeutet in diesem Fall eine Bildwiederholungsrate von mindestens 25 Bildern pro Sekunde, ein sensorisches haptisches Feedback muss mit 1000 Hz erfolgen.

Einen speziellen Anwendungsfall stellen medizinische Simulatoren für die chirurgische Ausbildung dar. Chirurgie ist Handarbeit, die nur mit viel Übung und Erfahrung zu guten Ergebnissen führt. In der klassischen Ausbildung erhält der angehende Chirurg diese an Präparaten oder durch Beobachtung eines erfahrenen Kollegen, der ihm dann einfache Handgriffe überlässt. Dies ist ein langwieriger und kostenintensiver Prozess.

Um eine effizientere Ausbildung zu ermöglichen, wurde deshalb nach Alternativen gesucht. Simulatoren sind solch eine Alternative. Die frühen Trainingssimulatoren waren einfache mechanische oder tierische Modelle. Mit Einführung der endoskopischen und vor allem der minimal-invasiven (knopflochchirurgischen) Eingriffe kamen Simulatoren vermehrt zum Einsatz. Da mechanische Modelle schnell an ihre Grenzen stoßen und tierische Modelle ethisch sehr umstritten sind, wurde mit der Entwicklung von computergestützten Simulatoren begonnen.

Computergestützte Simulatoren bieten wesentliche Vorteile gegenüber der konventionellen Ausbildung. Es entsteht keinerlei Risiko durch Training am Patienten, es bietet sich die Möglichkeit des Trainings selten vorkommender Pathologien und die einfache Wiederholbarkeit des Eingriffes. Es steht damit auch eine objektive Bewertung zur Verfügung, da alle Prüflinge mit derselben Aufgabe konfrontiert werden können. Ebenso ergibt sich daraus die Möglichkeit der Standardisierung der Ausbildung.

Die Entwicklung computergestützter medizinischer Trainingssimulatoren ist immer noch eine recht junge Disziplin. In den letzten Jahren wurde viel Grundlagenarbeit geleistet. Dabei haben sich einige grundsätzliche Aspekte herauskristallisiert, die medizinische Simulatoren erfüllen sollten. Dazu gehören geeignete Interfaces (Schnittstellen), Weichgewebesimulation, Kollisionserkennung und -antwort. Einen Überblick über diese Schlüsseltechnologien gibt die Arbeit von Grimm [32]. Auf diesen Forschungsgebieten ist immer noch viel Arbeit zu leisten, jedoch können mit den heute schon verfügbaren Methoden gute Ergebnisse erzielt werden.

Damit sich medizinische Simulatoren zur Ausbildung einsetzen lassen, genügt es nicht, eine vollkommene virtuelle Umgebung zu schaffen, sie müssen auch pädagogischen Ansprüchen gerecht werden. Pädagogisch relevante Aspekte sind ein Curriculum und ein Bewertungssystem.

Es gibt eine Reihe von Einsatzmöglichkeiten für medizinische Trainingssimulatoren, diese sind z.B.: die Arthroskopie (Kniespiegelung), Koloskopie (Darmspiegelung), Laparoskopie (Spiegelung der Bauchhöhle), neurochirurgische Eingriffe und ophthalmologische Eingriffe.

Der vermehrte Einsatz computergestützter medizinischer Trainingssimulatoren in der medizinischen Aus- und Fortbildung führt zu einer raschen Entwicklung auf dem Gebiet der Trainingssimulatoren.

### 1.3 Fragestellung

Der Zweck dieser Arbeit bestand darin, Algorithmen und Anwendungen zu entwickeln, die eine realistische Simulation der Phakoemulsifikation zur medizinischen Ausbildung ermöglichen. Die Phakoemulsifikation ist eine Methode der Kataraktoperation, für die spezielle maschinengesteuerte Instrumente benötigt werden. Die Entwicklung und Modellierung einer virtuellen Operationsmaschine, welche die reale Operationsmaschine nachahmt, bildete den Hauptteil dieser Arbeit. Weitere Modelle wurden für das Druck- und Volumenverhalten der Vorderkammer während der Operation erarbeitet. Neben diesen Aufgaben wurden Modelle für das Gewebeverhalten während der virtuellen Operation entwickelt.

Die Modelle wurden als Erweiterungen für den vorhandenen Augenoperationssimulator Eyesi implementiert. Bei der Entwicklung wurde darauf geachtet, dass die Schnittstellen zwischen Eyesi und der virtuellen Operationsmaschine klar strukturiert sind, damit Eyesi leicht um weitere virtuelle Operationsmaschinen erweiterbar ist. Dies soll ermöglichen, dass angehende Chirurgen an verschiedenen Maschinen trainieren können und somit ein Gefühl für eine spezielle Maschine erlangen können, bevor sie diese in der Realität einsetzen.

Eine spezielle Anwendung der Modelle stellen die angefertigten Trainingsmodule zur Simulation der Phakoemulsifikation dar.

### 1.4 Überblick über die Arbeit

Kapitel 2 gibt einen Einblick in die medizinischen Grundlagen, die für das Verständnis der Arbeit benötigt werden. Es wird das Krankheitsbild Katarakt und dessen Operation behandelt, sowie die damit einhergehenden Anforderungen an die Ausbildung.

In Kapitel 3 werden die Grundlagen der Simulation behandelt. Das Kapitel beinhaltet Informationen zur Modellbildung, numerischen Integration und Kollisionsbehandlung. Dabei werden Simulationsmodelle für Festkörper, deformierbare Körper und Flüssigkeiten vorgestellt. Neben den Grundlagen werden die für die später vorgestellten Trainingsmodule wichtigen Teile behandelt.

Kapitel 4 bietet einen Überblick über die aktuellen Trainingssimulatoren für die Kataraktoperation. Ein größerer Abschnitt wird Eyesi, das die Basis dieser Arbeit bildet, gewidmet.

Kapitel 5 stellt den Hauptteil der Arbeit dar, die Entwicklung einer virtuellen Operationsmaschine für die Phakoemulsifikation. Es wird ein Modell

vorgestellt, das alle relevanten Eigenschaften abbildet und dennoch flexibel ist. Bei der Modellbildung wurde auf die Erweiterbarkeit des Modells geachtet, um neue reale Entwicklungen in Zukunft leicht integrieren zu können. Das entwickelte Modell wurde mit Messwerten an einer realen Maschine und Werten aus der Theorie verglichen.

In Kapitel 6 werden Modelle für die Gewebeinteraktionen mit dem Instrumentarium entwickelt. Großer Wert wird auf den Einfluss durch die Operationsmaschine gelegt.

In Kapitel 7 werden die Eyesi-Software-Erweiterungen erläutert. Dies sind die Integration der virtuellen Operationsmaschinen und das Druck- und Volumenverhalten des Auges. Als zusätzliche Software wurde der Maschinentester entwickelt. Dieser ist eine Testumgebung für die virtuelle Operationsmaschine und das virtuelle Auge.

In Kapitel 8 werden die während der Arbeit entwickelten Trainingsmodule vorgestellt. Dies sind ein Prototyp der Phakoemulsifikation, ein abstraktes Trainingsmodul und die Simulation der Phakoemulsifikation. Im abstrakten Trainingsmodul soll speziell der Umgang mit der Operationsmaschine gelernt werden. Die Simulation der Phakoemulsifikation ist die Kombination aller zuvor eingeführten Teilbereiche.

Abschließend folgt in Kapitel 9 die Diskussion der entwickelten Modelle und ein Ausblick.





## Medizinischer Hintergrund

In diesem Kapitel werden die medizinischen Grundlagen beschrieben, die für das Verständnis dieser Arbeit notwendig sind. Dazu gehören die Anatomie des Auges [97] und [78], die Beschreibung des Krankheitsbildes Katarakt, ein historischer Überblick über die Kataraktoperation [79] und eine genauere Beschreibung der aktuellen Operationsmethode der Katarakt, sowie der benötigten Operationsmaschine [16], [80] und [4].

### 2.1 Der Augapfel

Der Augapfel (*Bulbus*) (siehe Abbildung 2.1) des Menschen kann grob in drei Abschnitte, drei Schichten und drei Räume eingeteilt werden. Die drei Abschnitte sind der vordere Abschnitt, der Übergangsbereich und der hintere Abschnitt. Die drei Schichten sind die Lederhaut (*Sclera*), die Aderhaut (*Choroidea*) und die Netzhaut (*Retina*). Im vorderen Abschnitt geht die Lederhaut in die lichtdurchlässige Hornhaut (*Cornea*) über. Die drei Räume sind die vordere Augenkammer, die hintere Augenkammer und der Glaskörper.

Der Augapfel hat einen ungefähren Durchmesser von 24 mm und in ihm herrscht, bei einem gesunden Menschen, ein Augeninnendruck von 10-22 mmHg.

**Nähere Betrachtung des vorderen Abschnitts** Der vordere Augenabschnitt (Abb. 2.1) ist durch die Hornhaut begrenzt. Sie ist durchsichtig und umgibt die vordere Augenkammer, in der sich die Iris befindet. Diese trennt die vordere von der hinteren Augenkammer. Die Iris dient als Blende und bedeckt teilweise die Linse. Die Linse ist über die Zonulafasern am Ziliarkörper befestigt und dort liegt der Ziliarmuskel. Die Brechkraft der Linse wird

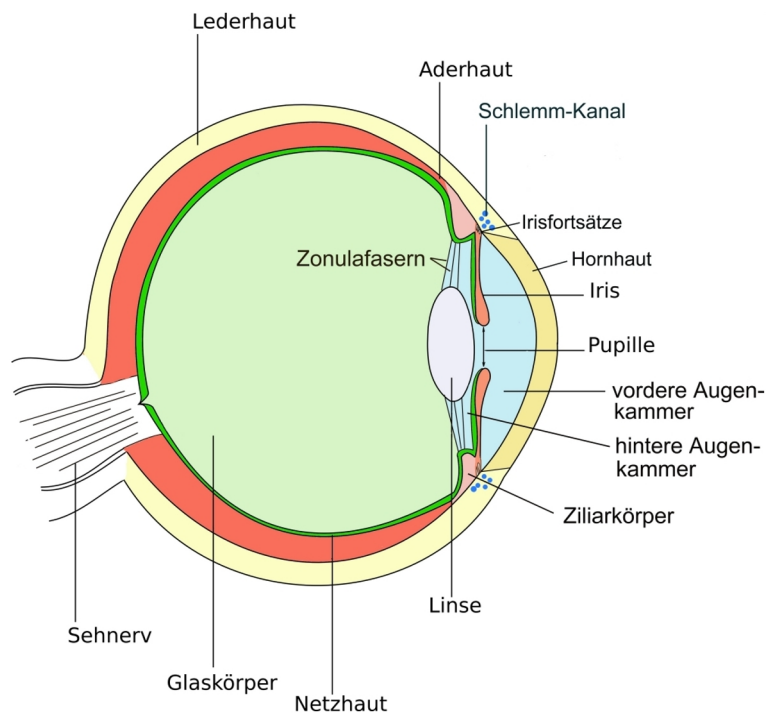


Abbildung 2.1: Der Querschnitt durch ein schematisches Auge (Quelle: Wikimedia Commons [90]).

durch die Kontraktion des Ziliarmuskels verändert. Dabei wird die Dicke der Linse beeinflusst. Dieser Vorgang wird als Akkommodation bezeichnet.

Die Linse (Abb. 2.2) besteht ihrerseits wieder aus drei Schichten, von außen nach innen sind dies der Kapselsack (oder kurz die Kapsel), der *Cortex* (Rinde) und der *Nucleus* (innere Kern). Die Linse wächst von innen nach außen und ist im Kern am härtesten.

Die Linse besitzt bei der Geburt einen Durchmesser von ca. 4,8 mm und wächst beim Erwachsenen auf ca. 10,0 mm. Die Dicke beträgt ca. 3,2 mm bei Geburt und 4,3 mm beim Erwachsenen. Die Dicke der Kapsel beträgt 2-4  $\mu\text{m}$ . Das Volumen des vorderen Abschnitts beträgt 0,15-0,35 ml. Der Abstand von der Linse zur Cornea beträgt ca. 3 mm.

## 2.2 Die Katarakt

Bei der Katarakt (siehe Abb. 2.3) auch Grauer Star genannt, trübt sich die sonst klare Linse. Der Begriff Katarakt bedeutet Wasserfall und stammt aus dem Mittelalter. Er behielt nach seiner lateinischen Wortherkunft das grammatikalische weibliche Geschlecht. Die damalige Vorstellung war, dass Flüssigkeiten, die vor der Linse herablaufen, erstarren und einen undurchsichtigen

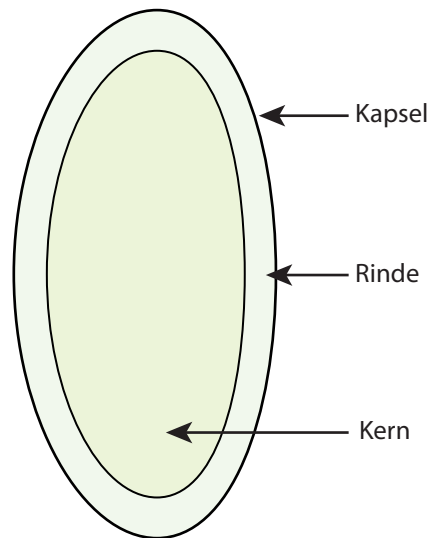


Abbildung 2.2: Schematische Darstellung der Schichten der Linse.

Film bilden. Seit dem 17. Jahrhundert weiß man, dass die Katarakt eine Erkrankung der Linse ist. Durch die Trübung wird das Sichtfeld eingeschränkt und bei Nichtbehandlung kommt es unweigerlich zur Erblindung. Die Katarakt ist meistens keine Erkrankung im ursprünglichen Sinn, sondern eher eine natürliche Ursache der Physiologie der Linse. Da die Linse zeitlebens weiter wächst, kommt es im hohen Alter zu einer immer stärkeren Verdichtung der Linse. Dies führt zu der Bildung der am häufigsten vorkommenden Variante, dem Alters-Star. Nur ein kleiner Teil der Katarakterkrankungen ist auf andere Ursachen, wie z. B. Masern, Diabetes mellitus, Medikamente, Drogen oder Augenverletzungen, zurückzuführen.

Die einzige Therapiemöglichkeit ist die Kataraktoperation, dabei wird die getrübbte Linse aus dem Auge entfernt und durch eine Kunstlinse ersetzt.

## 2.3 Die Kataraktoperation

Im Laufe der Zeit haben sich verschiedene Arten der Behandlung entwickelt. Sie haben jedoch eines gemeinsam: die getrübbte Linse muss entfernt werden. Bevor die heute gebräuchlichste Methode näher erläutert wird, soll ein kurzer Überblick über die Geschichte der Kataraktoperation vermittelt werden.

Die älteste Methode ist das Starstechen, sie wurde schon um ca. 800-600 v. Chr. in Indien und Persien praktiziert. Mit einem lanzenähnlichen Instrument, der Starnadel, wird dabei die Linse nach hinten in den Glaskörper

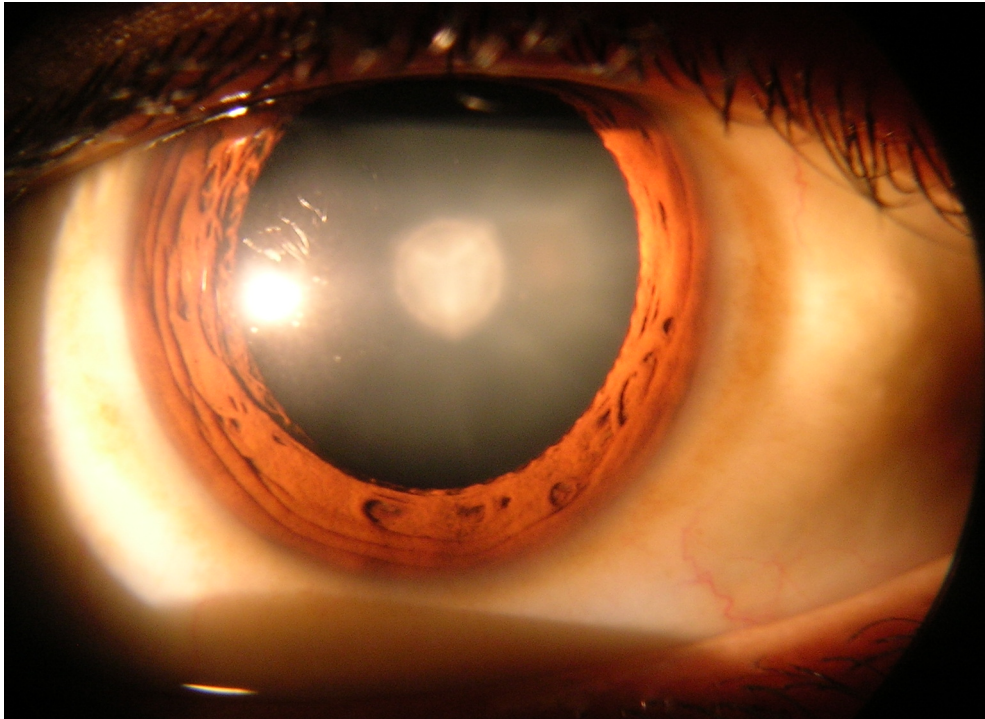


Abbildung 2.3: Die Katarakt (Quelle: Wikimedia Commons [3]).

gedrückt, so dass sie nach unten sinkt und den Lichteinfall ins Auge nicht mehr behindert. Nach dem Eingriff können die Patienten zwar wieder sehen, jedoch nur unscharf, da durch die fehlende Linse keine Möglichkeit der Akkommodation mehr gegeben war. Bei dieser Art der Operation bestand ein hohes Risiko der Infektion, da die verwendeten Operationsinstrumente selten gereinigt und nie sterilisiert wurden. So durfte die Linsenkapsel nicht verletzt werden, da es sonst zu einer Trübung des gesamten Glaskörpers kam.

Die Methode wurde durch die Araber ca. 850 n. Chr. durch ein zusätzliches Absaugen erweitert. Diese Methode geriet bis ins 19. Jahrhundert wieder in Vergessenheit.

Im Jahre 1753 publizierte Jacques Daviel eine neuartige Methode der Kataraktoperation, diese entspricht der noch heute durchgeführten extrakapsulären Kataraktextraktion.

Die Erfindung der intraokularen Linse durch Harold Ridley im Jahre 1949 war ein weiterer Meilenstein der Kataraktchirurgie. Es dauerte einige Jahre bis sich die intraokularen Linsen durchgesetzt hatten und angemessen implantiert werden konnten.

Im Jahre 1961 stellte der Pole T. Krawicz seine Methode der intrakapsulären Kataraktextraktion vor. Die Linse wird dabei an einen Kältestift angefroren und komplett aus dem Auge entfernt.

Ab diesem Zeitpunkt wird zwischen der *extrakapsulären* (ECCE) und *intrakapsulären* Kataraktextraktion (ICCE) unterschieden.

Bei der extrakapsulären Methode wird die Linsenkapsel schlitzförmig eröffnet und der komplette Linsenkern aus dem Auge entfernt. Der Kapselsack verbleibt als Halterung für die künstliche Linse im Auge und bildet weiterhin eine Trennung zum Glaskörper. Allerdings wird ein relativ großer Schnitt benötigt. Dieser muss mit einer Naht wieder verschlossen werden. Diese Naht führt zu einer stärkeren Hornhautkrümmung, die sich nach einigen Wochen bis Monaten normalisiert.

Bei der intrakapsulären Variante wird die Linse mitsamt dem Kapselsack aus dem Auge entfernt. Danach wird eine Vorderkammerlinse eingesetzt, die an der Iris angenäht wird. Um den Zugang zum Operationsgebiet zu ermöglichen, ist ein relativ großer Schnitt am Übergang der Lederhaut zur Hornhaut nötig. Der Vorteil dieser Variante ist der geringe Operationsaufwand. Die Nachteile überwiegen jedoch. Nach der Entfernung besteht keine Trennung mehr zwischen dem hinteren und dem vorderen Bereich, dadurch wird die Stabilität des Glaskörpers gestört und die Wahrscheinlichkeit einer Netzhautablösung steigt. Auch können nun leichter entzündungsfördernde Substanzen in den Glaskörper gelangen und eine entzündliche Reaktion der Netzhaut verursachen (cystoides Maculaödem).

1967 entwickelte Charles D. Kelman die *Phakoemulsifikation*, die Entfernung der Linse mit einem Ultraschall-Saug-Spül-Gerät. Dies ermöglichte den Zugang über wesentlich kleinere Schnitte im Auge. Die Regenerationszeit wurde dadurch deutlich kürzer.

### 2.3.1 Phakoemulsifikation

Entwickelt wurde die Phakoemulsifikation von Charles D. Kelman. Er hatte sich zum Ziel gesetzt, die Linse durch eine wesentlich kleinere Öffnung als bis zu diesem Zeitpunkt möglich aus dem Auge zu entfernen. Nachdem sämtliche Versuche fehlgeschlagen waren, kam ihm bei einem Zahnarztbesuch die Idee Ultraschall einzusetzen, um die Linse aus dem Auge zu entfernen. Das erste Ultraschallhandstück war auch ein modifiziertes zahnärztliches Instrument.

1967 wurde dann die erste Phakoemulsifikation am Patienten durchgeführt. Seit dieser Zeit stieg die Zahl der durch Phakoemulsifikation entfernten Katarakte stetig an und hat fast alle anderen Methoden verdrängt. Bei der Phakoemulsifikation werden wesentlich kleinere, selbst verschließende Schnitte und nur eine lokale Anästhesie benötigt. Dadurch können die Patienten meist schon am ersten oder zweiten Tag ohne größere Probleme das Krankenhaus verlassen.

Die Operation besteht aus mehreren Schritten: die Vorbereitungen, die Entfernung der Linse und das Einsetzen der Kunstlinse. Zu den Vorbereitungen zählt das Setzen der Schnitte als Zugang zum Situs (Operationsgebiet). Es wird ein Hauptschnitt für das Phakohandstück und zusätzliche ein bis zwei kleinere Schnitte für weitere Instrumente benötigt. Als nächstes wird die Vorderkammer mit *Viskoelastikum*<sup>1</sup> gefüllt. Dies dient zum einen als Schutz der Cornea vor dem Ultraschall und zum anderen wird damit eine gewisse Stabilität der Vorderkammer erreicht. Danach beginnt mit der *Kapsulorhexis* die eigentliche Prozedur. Die Kapsulorhexis ist eine kreisrunde Eröffnung der Linsenkapsel. Der Chirurg benutzt entweder eine spezielle Pinzette oder ein *Cystotome*<sup>2</sup>. Ist die Kapsulorhexis erfolgreich abgeschlossen, kann mit der Phakoemulsifikation begonnen werden.

Über die Jahre wurden verschiedene Phakoemulsifikationstechniken entwickelt, wie z. B.:

- Horizontal Chop,
- Vertical Chop,
- Chop and Stop und
- Divide and Conquer.

Gemeinsam haben alle diese Methoden, dass die Linse mittels Ultraschall verflüssigt und abgesaugt wird. Die später vorgestellte Simulation der Phakoemulsifikation lehnt sich an die Divide and Conquer Methode an.

**Divide and Conquer** Bei der Divide and Conquer Methode wird zunächst mittels Ultraschall ein Graben in die Linse gezogen und die Linse an dieser Stelle auseinander gebrochen. Die Linse wird dann um 90° rotiert und der Vorgang wird wiederholt. Die so entstandenen vier Teilstücke werden mit dem Phakohandstück angesaugt und in der Mitte der Vorderkammer verflüssigt und abgesaugt.

Die Kapselpolitur mittels *Irrigation und Aspiration* schließt die Linsenentfernung ab. Anschließend wird eine Kunstlinse eingesetzt und die Operation ist beendet.

### 2.3.2 Das Instrumentarium

Die Phakoemulsifikation ist eine Operationsmethode unter der Zuhilfenahme von aktiven Instrumenten. Dies bedeutet, dass es Eigenschaften der Instru-

---

<sup>1</sup> Eine hochviskose Flüssigkeit.

<sup>2</sup> Eine abgewinkelte Kanüle, mit der die Kapsel geschnitten bzw. gerissen werden kann.

mente gibt, die über eine *Operationsmaschine* gesteuert werden. Das Gelingen der Operation hängt stark vom Umgang mit der Operationsmaschine ab. Das Phakohandstück besteht aus einer Hohnadel, die zum Schwingen angeregt wird. Durch den Lumen wird das verflüssigte Linsengewebe aus der Vorderkammer entfernt. Des Weiteren muss ein Zufluss an Flüssigkeit gewährleistet sein, um den Druck in der Vorderkammer aufrecht zu erhalten.

Grundsätzlich sind folgende Werte über ein Fußpedal an der Maschine zu steuern:

- die Irrigation,
- die Aspiration und
- die Ultraschallstärke.

Die Irrigation ist für den Zufluss der Salzlösung in die Vorderkammer verantwortlich, sie wirkt sich auf die Kammerstabilität aus. Die Irrigationsstärke wird über die Irrigationsflaschenhöhe gesteuert. Diese wird in cm über dem Patientenkopf oder in entsprechendem Druck in mmHg angegeben. Die Aspiration bewirkt den Abtransport von Flüssigkeit und Linsengewebe. Die Aspiration kann über zwei Parameter eingestellt werden, den Durchfluss und das Vakuum. Der Durchfluss wird in ml/min eingestellt und ist für die Bewegung in der Vorderkammer verantwortlich. Je höher der Fluss ist, desto schneller bewegen sich die Linsestücke in der Vorderkammer. Das Vakuum wird in mmHg eingestellt. Über das Vakuum lässt sich die Anhaftung des Linsenmaterials an der Instrumentspitze einstellen. Die Aspiration ist eine aktive Instrumenteneigenschaft und wird durch eine Pumpe erzeugt. Der Ultraschall wird meist in  $\mu\text{m}$  oder % angegeben und steuert die Auslenkung der Phakonadel.

**Erweiterungen** Die Technik der Phakoemulsifikationsmaschinen wird immer noch weiter entwickelt. Hier sollen einige Neuerungen erwähnt werden.

Bei der *Torsional Phaco* schwingt das Instrument nicht vor und zurück, sondern bewegt sich seitlich. Dies verhindert die sonst vorhandene abstoßende Wirkung des Ultraschalls. Die Wärmeentwicklung am Schaft ist ebenfalls geringer.

Eine Weiterentwicklung sind die bimanuellen Handstücke. Bei den bimanuellen Instrumenten erfolgt die Irrigation durch ein anderes Handstück als die Aspiration, dies wurde erst durch die Entwicklung der *Cold Phaco* ermöglicht, da sich hierbei das Handstück fast nicht mehr erwärmt. Die Durchmesser dieser Instrumente sind wiederum kleiner als die der koaxialen. Die Zugänge am Auge sind dadurch kleiner und verheilen schneller.

Der derzeitige Trend geht zu immer kleineren Instrumenten. Dadurch soll die Belastung für den Patienten minimiert werden. Aus technischer Sicht ist ein Ende dieser Verkleinerungen abzusehen. Um die Belastung für den Patienten dennoch zu verringern, hilft nur eine fundierte Ausbildung des Chirurgen.

### 2.3.3 Anforderungen an die Ausbildung

Bei der Phakoemulsifikation sitzt der Operateur am Kopfende des liegenden Patienten. Er betrachtet dabei das zu operierende Gebiet durch ein Stereomikroskop. Während der Operateur in das Stereomikroskop blickt, nimmt er die Umgebung nur über das Gehör wahr. Durch die Audio-Ausgabe der Maschine erhält er ein Feedback über die Maschinenaktivität. Mit unterschiedlichen Fußpedalen kann er die Mikroskop- und Maschineneinstellungen manipulieren. Bei Unachtsamkeiten kann es leicht zu Verletzungen durch Ultraschall kommen. Eine weitere Komplikation ist der *Surge*-Effekt. Als Surge-Effekt wird ein durch erhöhten Aspirationsfluss erzeugter Druckabfall in der Vorderkammer bezeichnet. Dies erzeugt Kammerinstabilitäten, die bis zum Kammerkollaps führen können. Auch kann bei einem Surge die Iris und anderes umliegendes Gewebe in Mitleidenschaft gezogen werden. Die derzeitige Art der Ausbildung sieht zuerst ein Training an präparierten Schweineaugen vor. Im nächsten Schritt darf der angehende Chirurg leichte Teile der Operation ausführen, z. B. das Absaugen der zerteilten Linse.

Konzepte, die während der Ausbildung vermittelt werden sollten, sind die Handhabung der Instrumente, die Steuerung der Maschine, der damit verbundene angemessene Einsatz von Ultraschall, sowie die Erhaltung einer stabilen Vorderkammer.

Der bei der Handhabung der Instrumente zu erzielende Lerneffekt ist die getrennte Hand-Auge-Koordination. Der Chirurg sieht bei einer Kataraktoperation das Operationsumfeld stark vergrößert durch ein Stereomikroskop. Da die Instrumente durch den Zugangspunkt zum Auge zusätzlich gelagert sind, ist die Abstimmung der Handbewegung mit dem Gesehenen für Anfänger schwer in Einklang zu bringen und erfordert viel Übung. Genauso spürt der Chirurg alle auftretenden Kräfte nur indirekt über den Handgriff. Über einen verwertbaren Tastsinn verfügt er somit nicht. Die Steuerung der Maschine erfolgt durch Voreinstellungen der einzelnen Werte bzw. ihrer Grenzwerte und wird über ein Fußpedal gesteuert. Dabei muss der Chirurg ein Gefühl für die einzelnen Schaltunkte des Fußpedals bekommen. Die Fußpedale verfügen zusätzlich über Schalter, mit denen sich Einstellungen der Maschine ändern lassen. Darum muss der Chirurg im Umgang mit dem Fußpedal geübt sein, um in komplizierten Situation kein Fehlverhalten zu riskieren.



## Grundlagen der Simulation

In diesem Kapitel werden Grundlagen besprochen, die für das Verständnis von Simulationen bei medizinischen Trainingssimulatoren benötigt werden. Realitätsnähe steht bei der Entwicklung von computergestützten Trainingssimulatoren im Vordergrund. Sie wird durch geeignete Visualisierung und Simulation geschaffen und sorgt damit für die Immersion. Die Immersion wird durch die Interaktion mit dem Benutzer aufrechterhalten. Diese Interaktion beinhaltet eine Reaktion der virtuellen Realität auf die Eingaben des Benutzers. Bei medizinischen Trainingssimulatoren sind solche Reaktionen Deformation oder Destruktion der modellierten Umgebung. Dies wird durch eine zeitliche Änderung des Modells erzielt. Die eingesetzten Verfahren müssen dabei immer dem Aspekt der Echtzeitfähigkeit<sup>1</sup> gerecht werden.

Die Literatur zu physikalisch mechanischer Simulation ist breit gefächert. Seit ihren Anfängen in den 1980er Jahren haben sich eine Reihe unterschiedlicher Ansätze entwickelt. Ihre Ursprünge haben alle in der klassischen Mechanik und der Kontinuumsmechanik. Durch die Übertragung auf eine computergestützte Simulation verlieren die zu simulierenden Objekte ihre Kontinuität, da Computer nur Daten verarbeiten können, die in diskreter Art vorliegen. Zur Beschreibung von Objekten werden Position, Geschwindigkeit und Beschleunigung verwendet. Die auf das Objekt wirkenden Kräfte haben dann je nach *Simulationsmodell* Einfluss auf die Bewegung bzw. Deformation des Objektes. Durch numerische Integration der Bewegungsgleichungen lässt sich ausgehend vom aktuellen Zeitpunkt  $t$  der Zustand des Objektes im nächsten Zeitpunkt  $t + \Delta t$ , mit dem Integrationszeitschritt  $\Delta t$ , bestimmen. Ist der Integrationszeitschritt  $\Delta t$  klein genug gewählt, kann ein für den Benutzer kontinuierliches Verhalten erzeugt werden. Der Zeitschritt  $\Delta t$  ist unter anderem abhängig von dem eingesetzten Integrationsverfahren. Bewegungen von Ob-

---

<sup>1</sup>Echtzeitfähigkeit bedeutet in diesem Fall, dass das System innerhalb einer für den Menschen nicht wahrnehmbaren Zeitspanne auf die Eingaben des Benutzers plausibel reagiert.

jekten können zu Überschneidungen dieser führen. Diese werden durch die *Kollisionserkennung* festgestellt und über die *Kollisionsantwort* aufgelöst.

Dies spiegelt den **generellen Ablauf der Simulation** wider. Es müssen folgenden Schritte durchgeführt werden:

- Computergrafische Darstellung der Szene zum Zeitpunkt  $t$ ,
- Kollisionsbehandlung,
- Berechnung aller auf das Objekte wirkenden Kräfte,
- Zeitliche numerische Integration zum nächsten Zeitpunkt  $t + \Delta t$ .

Im Folgenden werden die einzelnen Punkte genauer betrachtet.

Bevor die dynamischen Vorgänge genauer untersucht werden, wird der allgemeine Aufbau einer virtuellen Szene beschrieben.

Im Unterkapitel Modellbildung wird erläutert, welche Möglichkeiten bestehen, einen realen Prozess abzubilden. Die Grundkonzepte der dynamischen physikalischen Simulation werden danach anhand von Partikelbewegungen erläutert. Anschließend werden einige Simulationsmodelle vorgestellt, die teilweise für die Entwicklung der Trainingsmodule benötigt wurden.

Abgeschlossen wird das Kapitel mit den Grundlagen über die Kollisionsbehandlung.

## 3.1 Szene

Als Szene wird im Folgenden der Teil der virtuellen Realität bezeichnet, der von der Grafikkarte gerastert und an ein Wiedergabegerät (Display) weitergeleitet wird. Zur Beschreibung einer solchen Szene stehen dem Entwickler verschiedene Möglichkeiten zur Verfügung. Eine weit verbreitete Variante ist das Unterteilen der Objekte in Polygone, bzw. Dreiecksgitter, die dann von der Grafikkarte gerastert werden. Es wird meist auf vorhandene Grafikbibliotheken wie OpenGL oder Direkt3D zurückgegriffen.

Die Grafikbibliotheken bieten nur einen rudimentären Umfang an Befehlen, die sich nicht zur Verwaltung von Objekten in der Szene eignen. Dafür wird ein so genannter Szenegraph verwendet. Ein Szenegraph ist ein gerichteter azyklischer Graph. Mit ihm lässt sich eine räumliche Anordnung von Objekten in einer logischen Struktur abbilden. Die Verwendung von Szenegraphen erlaubt es Objektpositionen und -orientierungen durch Koordinatentransformationen darzustellen. Dabei wird zwischen Objektkoordinaten- und Weltkoordinatensystemen unterschieden. Die Position des im Objektkoordinatensystem vorliegenden Dreiecks im Weltkoordinatensystem ergibt sich durch

hintereinander Ausführung von Matrixoperationen. Auf diese Weise können Objektkoordinaten von einem System in ein anderes Objektkoordinatensystem durchgeführt werden.

Für weitere Informationen wird auf die quelloffene Implementierung Openscenegraph<sup>2</sup> eines Szenegraphen verwiesen. In der VRM-Bibliothek ist solch ein Szenegraph zur Objektverwaltung implementiert. Die Visualisierung basiert auf OpenGL.

## 3.2 Modellbildung

Von Anfang an wurde in der Physik versucht mittels mathematischer Hilfe die Natur um uns herum zu beschreiben. Dafür wurden verschiedenste Modelle entwickelt, verworfen, neu entwickelt und verbessert. Schill bezeichnet dies als den *modeling circle* [76]. Die meisten Modelle liefern für eine bestimmte Situation eine ausreichend genaue Beschreibung, können aber nicht universell eingesetzt werden. So wurden in den vergangenen Jahren zahlreiche physikalisch basierte Modelle für Festkörper, deformierbare Objekte und Flüssigkeiten entwickelt. Nach den Grundlagen der Dynamik werden diese Modelle näher erläutert. Die in dieser Arbeit verwendeten Modelle werden etwas ausführlicher behandelt. Einen umfangreichen Überblick über die in der VRM-Bibliothek implementierten Modelle liefert [32].

**Partikel und Partikelsysteme** Eine in der Echtzeitsimulation weitverbreitete Methode Objekte zu simulieren, sind partikelbasierte Systeme.

Ein Partikel  $P$  ist ein dimensionsloser Massenpunkt mit der Masse  $m$  und folgenden zeitabhängigen Eigenschaften: einem Positionsvektor  $\vec{r}(t)$  in Bezug auf ein Referenzkoordinatensystem, einem Geschwindigkeitsvektor  $\vec{v}(t)$  sowie einen Beschleunigungsvektor  $\vec{a}(t)$ .

Kombiniert man  $n$  Partikel  $P$ , so erhält man ein *Partikelsystem*. Man fasst die einzelnen Komponenten für Position, Geschwindigkeit und Beschleunigung zu jeweils  $3n$ -dimensionalen Vektoren  $\mathbf{x}, \mathbf{v}, \mathbf{a}$  zusammen. Das Simulationsmodell bestimmt dabei das Verhalten der Partikel über innere Kräfte. Für die Kraft  $\vec{F}$  auf ein Partikel  $i$  gilt dann:

$$\vec{F}_i(\mathbf{x}, \mathbf{v}) = \vec{H}_i(\mathbf{x}, \mathbf{v}) + \vec{D}_i(\vec{v}_i) + \vec{E}_i \quad (3.1)$$

wobei  $\vec{H}_i(\mathbf{x}, \mathbf{v})$  die im Modell herrschenden inneren Kräfte,  $\vec{D}_i(\vec{v}_i)$  eine partikel- und geschwindigkeitsabhängige Dämpfung und  $\vec{E}_i$  externe äußere Kräfte auf

---

<sup>2</sup><http://www.openscenegraph.org/>

den Partikel bezeichnen. Für das Verhalten der Partikel untereinander ist das verwendete Modell  $\vec{H}_i(\mathbf{x}, \mathbf{v})$  ausschlaggebend, das hier ganz allgemein für Partikelpositionen und -geschwindigkeiten angegeben ist. In die meisten Modelle fließen nur die Partikelpositionen ein.

Die Modelle lassen sich grob in Simulationen von Flüssigkeiten bzw. Gasen, deformierbaren Körpern und Festkörpern unterteilen. Für die Modellbildung  $\vec{H}_i(\mathbf{x}, \mathbf{v})$  ist die Kopplung der Partikel untereinander entscheidend. Kopplung bedeutet hierbei die Einhaltung einer topologischen Vorgabe, die durch den Ursprungszustand gegeben ist. So herrscht bei Flüssigkeiten und Gasen keinerlei Kopplung der Partikel untereinander. Sie können sich frei im Raum bewegen. Im Zusammenhang mit deformierbaren Objekten wird oft von dem Begriff Gitter (eng. *Mesh*) Gebrauch gemacht. Dies verdeutlicht den topologisch festen Zusammenhalt der Partikel untereinander. Bei deformierbaren Objekten darf es zu Verformungen kommen, jedoch bleibt die Topologie ihres Ausgangszustands erhalten. Beim Modell der Festkörper werden die inneren Kräfte als so stark angenommen, dass es zu keinerlei Verschiebung untereinander kommt. Kräfte wirken sich immer nur auf das Verhalten des gesamten Objektes aus.

Um ein Verständnis für die einzelnen Modelle zu bekommen, sollen ihre Grundlagen hier vorgestellt werden.

#### 3.2.1 Festkörpersimulation

Ein einfaches, oft ausreichendes, Modell ist das der Festkörper. Die Festkörpersimulation gehört zu den ältesten Simulationen, sie wurden bereits Mitte der 1980er Jahre erwähnt [53] und [34]. Sie basieren auf den Grundlagen der *Mechanik starrer Körper*. Einen umfangreichen Überblick bieten die Ausführungen von Baraff und Witkin auf der Siggraph 1997 [8]. In der Fachliteratur wird dieses Thema ausführlich behandelt. Es sei auf Eberly [20] verwiesen. Betrachtet man das Modell aus der Sicht der inneren Kräfte, so lässt das Modell des Festkörpers keine inneren Verschiebungen zu. Deshalb ist es nicht notwendig, innere Kräfte  $\vec{H}_i(\mathbf{x}, \mathbf{v})$  zu berechnen. Ein Festkörper ist vollkommen durch seinen Schwerpunkt und seine Massenverteilung repräsentiert. Bewegungen von Festkörpern lassen sich durch einen linearen und einen angularen Anteil beschreiben. Der lineare Anteil wird durch einen Vektor  $\vec{x}_s(t)$ , der Position des Schwerpunkts und der angular durch eine Matrix  $R(t)$  für die Orientierung abhängig von der Zeit  $t$  angegeben. Äußere Kräfte, die an Festkörpern angreifen, wirken somit auf die lineare und angular Beschleunigung. Die zeitliche Integration dieser Größen führt zu einer Änderung der Position des Schwerpunktes und der Orientierung des Festkörpers im Raum. Fasst man Festkörper, ähnlich wie Partikel, zusammen, erhält man ein *Mehrkörpersystem* (MKS). Analog zu Gleichung 3.1 können

Modelle für die Kräfte, die unter den Körpern wirken, entwickelt werden. So verwenden Baraff und Witkin [8] eine Reihe von Modellen, die auf dem Prinzip der neutralen Energie basieren. Damit ist es möglich, Verbindungen (wie Gelenke (engl. *joints*) oder Drehachsen (engl. *hinges*)) und Kontakte zu modellieren. Der zeitlich beschränkende Faktor bei Mehrkörpersystemen ist die Kollisionserkennung und Kollisionsauflösung. In neueren Arbeiten werden Echtzeitsimulationen mit mehreren hundert bis tausend Objekten vorgestellt [59] und [44]. Im Abschnitt 3.4 - Kollisionsbehandlung wird genauer auf das Problem der Berechnung der Kontaktkräfte eingegangen. Weitere Probleme sind Zittereffekte [35] durch die Kollisionsantwort und die korrekte Reibungsberechnung [44].

### 3.2.2 Simulation deformierbarer Objekte

Die Literatur zum Thema Simulation deformierbarer Objekte ist sehr umfangreich und es haben sich im Laufe der Zeit viele unterschiedliche Ansätze entwickelt. Die Ansätze haben jedoch eine Gemeinsamkeit. Es werden immer interne Kräfte berechnet, die zu einer Verformung des Objektes führen. Bei traditionellen Lagrange-Ansätzen werden deformierbare Objekte, wie in Gleichung 3.1, durch Partikelsysteme beschrieben. Über die Randbedingungen  $\vec{C}$  (engl. *Constraints*) wird dabei das Verhalten der Partikel festgelegt. Einen Überblick über die verschiedenen Ansätze bieten [26] und [67]. Nealen et al. [67] unterteilen die Lagrange Ansätzen weiter in gitterbasierte (engl. *mesh-based*) und gitterfreie (engl. *mesh-free*) Ansätze.

Die Grundlage vieler Deformationsmodelle ist die Kontinuumsmechanik, diese beschäftigt sich mit dem Verhältnis vom undeformierten Objekt zum deformierten. Dies lässt sich über Verzerrungstensoren und Spannungstensoren ermitteln.

#### 3.2.2.1 Finite Elemente Methode - FEM

Bei der Finite Elemente Methode handelt es sich um ein numerisches Verfahren zur Lösung von Differentialgleichungen. Die zu lösenden Differentialgleichungen für deformierbare Objekte sind durch die Elastizitätstheorie (Kontinuumsmechanik) gegeben. Dabei wird das zu untersuchende Objekt in Finite (endliche) Elemente unterteilt. Für jedes Finite Elemente wird eine kontinuierliche *Ansatzfunktion* definiert. Die Ansatzfunktionen werden zu einem Gleichungssystem zusammengesetzt, das numerisch gelöst werden kann. Die Lösung liefert die Deformation des Gesamtobjektes.

Die Verwendung der nichtlinearen Kontinuumsmechanik führt zu nichtlinearen Differentialgleichungen. Die FEM liefert dafür sehr genaue Lösungen, ist

aber sehr rechenintensiv. Deshalb wird oft auf linearisierte elastische Modelle zurückgegriffen [13]. Durch die Linearisierung verlieren die Finiten Elemente ihre Rotationsinvarianz. Dies führt bei starken Deformationen zu Artefakten. Zur Lösung dieses Problems werden Hybrid-Ansätze, die Festkörper und FEM kombinieren, verwendet. Müller et al. [62] rechnen die Rotation aus dem Simulationsobjekt heraus und wenden die Kräfte auf das unrotierte Objekt an.

**Verwandte Ansätze** Als weitere gitterbasierte Ansätze können die Finite Differenzen Methode (FDM), Finite Volumen Methode (FVM) und Boundary Element Method (BEM) genannt werden. Dabei wird von speziellen Annahmen ausgegangen, um die Berechnungen zu beschleunigen. So wird bei den Finite Differenzen auf eine kontinuierliche Ansatzfunktion verzichtet und durch Differenzen angenähert. Dies lässt sich nur auf regulären Gitter verwenden, dort erleichtert es allerdings die Implementierung.

Im Gegensatz zur FEM, die nur Knotenpositionen der Ansatzfunktion verwendet, bezieht sich die Finite Volumen Methode auf die gesamte Fläche eines Elementes. Dies spart einerseits Rechenzeit und gibt einen direkteren Zugang zum Modellverständnis.

Eine alternative Vorgehensweise stellt die Boundary Element Method dar, bei ihr werden die Randelemente (engl. *Boundary Elements*) zur Berechnung der Deformation herangezogen. Dadurch erhält man ein einfacher zu lösendes zweidimensionales Problem. Der Nachteil dieser Vereinfachung ist, dass nur homogene Körper simuliert werden können [43].

#### 3.2.2.2 Feder-Masse-Systeme

Feder-Masse-Systeme tauchen schon in den 1980ern in der Literatur auf [72] und [100]. Sie sind intuitiv und einfach zu implementieren. Wie der Name schon sagt, besteht das Modell aus einzelnen Massepunkten, auch Knoten genannt, die über Federn verbunden sind. Sie sind somit eine spezielle Art von Partikelsystemen, wobei zwei Partikel fest über eine Feder verbunden sind. Für die (virtuellen) Federn gilt das Hooke'sche Gesetz (Gleichung 3.2) in seiner einfachsten Form:

$$\vec{F} = k\Delta l \quad (3.2)$$

mit der Auslenkung  $\Delta l$ , der Federkonstanten  $k$  und der resultierenden Kraft  $\vec{F}$  in Federrichtung. Dabei wird im elastischen Bereich des Materials nur ein linearer Zusammenhang zwischen Kraft und Federlänge beachtet, ohne auf die Querschnittsflächenänderung einzugehen. Im dreidimensionalen Raum ist

die Federrichtung durch den Verbindungsvektor zwischen den Knoten gegeben. Als Kraft zwischen zwei Masse-Punkten gilt:

$$\vec{F}_{ij} = -\frac{\vec{d}_{ij}}{\|\vec{d}_{ij}\|} k_{ij} (\|\vec{d}_{ij}\| - l_{ij}) \quad (3.3)$$

mit  $\vec{d}_{ij} = \vec{x}_i - \vec{x}_j$  und der Federnulllänge  $l_{ij}$ .

Verbindet man viele Knoten über virtuelle Federn so erhält man eine Gitterstruktur. Kommt es durch eine äußere Kraft zu einer Verschiebung von Knoten, ändert sich damit die Länge der Federn. Über das Hooke'sche Gesetz wird die Kraft berechnet, die durch die Feder auf die mit ihr verbundenen Knoten ausgeübt wird. Da für Kräfte das Superpositionsprinzip gilt, werden alle auf einen Knoten wirkenden Kräfte für den aktuellen Zeitschritt akkumuliert. Im Allgemeinen gilt für die auf einen Knoten  $i$  wirkende Kraft:

$$\vec{F}_i = \vec{F}_{i, \text{intern}} + \vec{F}_{i, \text{extern}} \quad (3.4)$$

wobei  $\vec{F}_{i, \text{intern}}$  die Feder-Kräfte sind und  $\vec{F}_{i, \text{extern}}$  Gravitations-, Kollisions- oder weitere äußere Kräfte sein können.

Über das 2. Newtonsche Gesetz ergibt sich aus der Kraft auf die Knoten die aktuelle Beschleunigung. Durch die numerische Integration (siehe Abschnitt 3.3) werden die neue Knotengeschwindigkeiten und -positionen berechnet. Die Verformung pflanzt sich so mit der Zeit fort, bis sich ein Gleichgewicht einstellt.

Eine bedeutende Rolle spielt in Feder-Masse-Systemen die Dämpfung. Sie hat zwei wesentliche Aufgaben:

1. Energiereduzierung: Ohne Dämpfung verliert ein virtuelles System keine Energie und es kommt zu keinem Abklingen der Schwingung.
2. Erhöhung der Stabilität: Numerische Integration ist fehlerbehaftet. Dies führt zu Instabilitäten, da das System immer mehr Energie aufnimmt, wenn sich die Rechenfehler aufsummieren. Die Dämpfung wirkt diesen Rechenfehlern entgegen. Dem System wird dadurch Energie entzogen.

Die Dämpfung wird direkt auf die Knotengeschwindigkeit  $\vec{v}$  ausgeübt, für die gedämpfte Geschwindigkeit  $\vec{v}_D$  gilt:

$$\vec{v}_D = k_D * \vec{v}, \quad k_D = [0, 1[ \quad (3.5)$$

wobei der Dämpfungskoeffizient  $k_D$  sorgfältig gewählt werden muss. Ist er zu groß, sieht das Verhalten unnatürlich aus. Ist er zu klein, kann das System zu leicht instabil werden. Zudem ist er abhängig von der Größe des Zeitschritts.

**Erweiterungen** Betrachtet man Feder-Masse-Systeme als Partikelsysteme, deren Partikel über eine weiche Randbedingung (engl. soft constraint), verknüpft sind, lässt sich eine Energie definieren:

$$E = \frac{1}{2}kC^2 \quad (3.6)$$

Wobei die Randbedingung  $C$  eine skalare Funktion der Knotenpositionen ist und die Abweichung vom undeformierten Zustand misst. Die Energie ist für den Ruhezustand des Objektes minimal. Durch Ableitung dieser Energie nach den Partikelpositionen

$$\vec{F}_i = \frac{\partial E}{\partial \vec{x}_i} \quad (3.7)$$

erhält man die Kraft  $\vec{F}_i$ , die nach einer Deformation die Partikel wieder in den Ruhezustand überführt. Eine einfache Abstandsrandbedingung entspricht dabei genau dem Feder-Masse-Modell.

Solche weichen Randbedingungen können für verschiedene Modellparameter verwendet werden. Teschner [94] verwendet diesen Ansatz für Flächen- (Gleichung 3.8) und Volumenrandbedingungen (Gleichung 3.9).

$$E_A(\vec{x}_i, \vec{x}_j, \vec{x}_k) = \frac{1}{2}k_A \left( \frac{\frac{1}{2} |(\vec{x}_j - \vec{x}_i) \times (\vec{x}_k - \vec{x}_i)| - A_0}{A_0} \right)^2 \quad (3.8)$$

Die Flächenrandbedingung basiert auf Dreiecken mit den Knotenpositionen  $\vec{x}_i, \vec{x}_j, \vec{x}_k$ , der Nullfläche  $A_0$  und der Flächensteifigkeit  $k_A$ .

$$E_V(\vec{x}_i, \vec{x}_j, \vec{x}_k, \vec{x}_l) = \frac{1}{2}k_V \left( \frac{\frac{1}{6}(\vec{x}_j - \vec{x}_i) \cdot ((\vec{x}_k - \vec{x}_i) \times (\vec{x}_l - \vec{x}_i)) - V_0}{V_0} \right)^2 \quad (3.9)$$

Die Volumenrandbedingung basiert auf Tetraedern mit den Knotenpositionen  $\vec{x}_i, \vec{x}_j, \vec{x}_k, \vec{x}_l$ , dem Nullvolumen  $V_0$  und der Volumensteifigkeit  $k_V$ .

Die Ableitung dieser Randbedingungen können somit als Flächen- und Volumenfedern bezeichnet werden. Diese ermöglichen eine volumenerhaltende Simulation. Das Modell wurde unter dem Namen *Deformation Energy Minimization* (DEM) in der VRM-Bibliothek implementiert [19], dabei wurde auf die von Teschner vorgeschlagene Normierung auf die Nullfläche  $A_0$  bzw. das Nullvolumen  $V_0$  verzichtet, da diese nach der Ableitung nicht mehr linear, sondern quadratisch in das Modell einfließt.



Neuere Erweiterungen beschäftigen sich mit der Modellierung nichtlinearen Verhaltens von Feder-Masse-Systemen. Dabei wird die Federkonstante  $k$  nicht mehr als konstant angenommen, man spricht von **Neo-Hooke'schem** Verhalten. Durch die Veränderung kann eine nichtlineare Kraftzunahme modelliert werden. Dies wird vor allem bei medizinischen Gewebesimulationen eingesetzt. Die Federkonstante kann dabei von anderen Parametern abhängig sein, z. B. von der Auslenkung oder der Temperatur. Teschner et al. verwenden in [92] und [93] einen quadratischen bzw. exponentiellen Ansatz zur Gewebesimulation des menschlichen Gesichts.

### 3.2.2.3 Verschiebungsorientierte Ansätze

Neben den physikalisch basierten Ansätzen gibt es verschiebungsorientierte Ansätze wie Chainmail [25], [77] und Dragnet [33]. Bei Chainmail ist das zu simulierende Objekt aus starren Kettengliedern, ähnlich einem mittelalterlichen Kettenhemd, modelliert. Die Kettenglieder können sich in einem gewissen Spielraum frei bewegen, bis sie an die Grenzen ihrer benachbarten Kettenglieder stoßen. Dabei wird die Bewegung auf den Nachbarn übertragen. Nach  $O(n)$  Schritten ist ein Gleichgewichtszustand hergestellt.

Beim Dragnet-Algorithmus sind die Knoten über so genannte *Strings* (Schnüre) verbunden. Jeder String hat eine maximale Länge, bis zu der er gedehnt werden kann. Wird diese Länge durch verschieben eines Knoten überschritten, wird die Stringlänge korrigiert. Dafür wird der mit ihm verbundenen noch nicht bewegte Knoten in Richtung des Strings verschoben. Der Algorithmus terminiert, wenn es keine Längenüberschreitungen mehr gibt. Dies ist spätestens der Fall, wenn alle Strings einmal korrigiert wurden.

### 3.2.2.4 Gitterfreie Methoden

Bei gitterfreien Methoden besteht keine topologische Beziehung zwischen den Knoten, die das deformierbare Objekt repräsentieren. Während der Simulation existiert keine feste Nachbarschaftsbeziehung, die ausgewertet werden könnte.

Gitterfreie Methoden können auf unterschiedliche Weise zur Simulation von deformierbaren Objekten eingesetzt werden. Ein Einsatzgebiet ist die Modellierung auf Basis der Kontinuumsmechanik. Dabei werden Ansatzfunktionen für die Umgebung von Partikel definiert. Diese Methode ist das Pendant zur FEM bei denen partielle Differentialgleichungen über die Topologie der Objektes bestimmt sind. Müller et al. [66] waren die ersten die solch einen Ansatz im Bereich der Computergrafik verwendeten.

Eine weitere Möglichkeit schlagen Müller et al. in [63] vor. Bei ihrem Shape-Matching-Verfahren wird der Ruhezustand des Objektes in jedem Zeitschritt über die Methode der kleinsten Quadrate an das deformierte Objekt angepasst. Die Knotenpositionen dieser angepassten Form dienen als Zielpositionen für die Partikel. Über ein entsprechendes Integrationsverfahren werden die Partikel auf die Zielpositionen gezogen.

### 3.2.3 Flüssigkeitssimulation

Neben Festkörpern und deformierbaren Körpern ist die Flüssigkeits- und Gassimulation ein weiteres Simulationsgebiet. Die Eigenschaften von Flüssigkeiten und Gasen unterscheiden sich wesentlich von denen der Festkörper und deformierbaren Objekten. Flüssigkeiten bilden zwar eine Oberfläche aus, besitzen jedoch keinen festen Zusammenhalt und können beliebig geteilt werden, es besteht somit keinerlei topologische Bindung unter den einzelnen Partikeln.

Flüssigkeiten werden durch die **Navier-Stokes-Gleichungen** beschrieben. Die Navier-Stokes-Gleichungen, die Grundgleichungen der Strömungsmechanik, koppeln die Moment- (Impuls-) und Massenerhaltung, welche nötig sind, um eine Flüssigkeitsbewegung vollständig zu beschreiben. Für Flüssigkeitssimulationen wird meist auf die Navier-Stokes-Gleichungen für inkompressible Flüssigkeiten zurückgegriffen:

$$\frac{\partial \vec{v}}{\partial t} + (\vec{v} \cdot \nabla) \vec{v} = -\nabla p + \nu \Delta \vec{v} + \vec{f} \quad (3.10)$$

$$\nabla \cdot \vec{v} = 0 \quad (3.11)$$

mit der Strömungsgeschwindigkeit  $\vec{v}$ , der Dichte  $\rho$ , der kinematischen Viskosität  $\nu = \eta/\rho$ , der dynamischen Viskosität  $\eta$  und der äußeren Kraft  $f$ . Die zweite Zeile ergibt sich aus der Divergenzfreiheit des Strömungsfeldes und führt zu der vereinfachten Kontinuitätsgleichung.

Um dieses System nichtlinearer partieller Differentialgleichungen zweiter Ordnung zu lösen, können die Euler- oder die Lagrange-Methode angewandt werden. Die Euler-Methode ist gitterbasiert und die Lagrange-Methode partikelbasiert. Diese werden nun näher untersucht.

#### 3.2.3.1 Gitterbasierte Ansätze

Bei gitterbasierten Ansätzen wird der zu untersuchende Raum in ein stationäres Gitter aufgeteilt. Jeder Gitterbereich besitzt dabei einen Wert für

die Strömungsgeschwindigkeit und die Dichte der Flüssigkeit. Während der Simulation verbleibt das Gitter am selben Ort, es ändern sich nur die Werte der verwendeten Variablen. Es wurde eine Reihe von Lösungsmöglichkeiten für die numerische Berechnung vorgeschlagen. Erwähnenswert in Bezug auf Echtzeitsimulationen sind die Arbeiten von Foster et al. [23] und Stam [88].

In [23] wird ein gitterbasierter Ansatz vorgestellt, der die Navier-Stokes-Gleichungen numerisch löst. Dabei wird der Raum in Quader aufgeteilt, wobei den Flächenmittelpunkten die Strömungsgeschwindigkeit und dem Volumenmittelpunkt die Druckverteilung zugewiesen wird. Die Navier-Stokes-Gleichungen werden dann mittels Finiter Differenzen gelöst.

Stam stellte in [88] einen echtzeitfähigen Ansatz vor. Durch einen so genannten „semi lagrangian“ Ansatz wird dabei selbst bei beliebig großen Zeitschritten die Stabilität gewährleistet. Durch eine Helmholtz-Hodge Dekomposition können die zu lösenden Gleichungen entkoppelt werden. Dies führt zu folgender Gleichung:

$$\frac{\partial \vec{v}}{\partial t} = P(-(\vec{v} \cdot \nabla) \vec{v} + \nu \Delta \vec{v} + \vec{f}) \quad (3.12)$$

mit der Projektion  $P$ , für die gilt:  $Pu = u$  und  $P\nabla p = 0$ . Die so entstandene Gleichung lässt sich in vier Teilschritte aufteilen, die unabhängig voneinander berechnet werden können. Die Teilschritte (in der Gleichung 3.12 von rechts nach links) bezeichnet Stam als *Body forces*, *Diffusion*, *Advection* und *Projection*. Dies wirkt sich positiv auf die Komplexität aus, da weniger Zeitschritte benötigt werden und erlaubt den Einsatz in Echtzeitsimulationen.

### 3.2.3.2 Partikelbasierte Ansätze

Bei partikelbasierten Ansätzen wird die Flüssigkeit als Vielzahl von Partikel modelliert, die sich frei im Raum bewegen können. Diese können untereinander gekoppelt oder ungekoppelt sein. Dabei bezieht sich Kopplung nicht auf eine topologische Verknüpfung, sondern auf eine Kraft übertragende Eigenschaft. So bestehen bei einem ungekoppelten System keine Wechselwirkungen zwischen den Partikeln, im Gegensatz dazu wirken bei einem gekoppelten die Partikel aufeinander. Ungekoppelte Systemen werden häufig nur für Grafikeffekte wie Fontänen oder Explosionen benutzt [69] und [74].

**Smoothed Particle Hydrodynamics** Ein weit verbreiteter Ansatz für ein gekoppeltes System ist Smoothed Particle Hydrodynamics (SPH). Ursprünglich wurde der SPH-Ansatz 1977 von Lucy [54] und Gingold und Monaghan [28] entwickelt, um astrophysikalische Phänomene wie Sternbildung und Kollisionen von Galaxien zu simulieren. Er kann jedoch auch für die Simulation

von Flüssigkeiten verwendet werden. In der Zwischenzeit ist SPH ein weit verbreiteter gitterfreier Ansatz. Die Grundidee ist ein kontinuierliches Feld anhand von diskreten Punkten zu definieren. Dazu wird eine radiale symmetrische Gewichtsfunktion eingesetzt, der *Smoothing Kernel*.

Bei der SPH wird ein quantitativer skalarer Wert  $A$  an der Stelle  $\vec{r}$  durch eine gewichtete Summe interpoliert, zu der alle Partikel beitragen:

$$A(\vec{r}) = \sum_j m_j \frac{A_j}{\rho_j} W(\vec{r} - \vec{r}_j, h) \quad (3.13)$$

wobei  $j$  über alle Partikel iteriert,  $m_j$  die Masse,  $\rho_j$  die Dichte,  $A_j$  das quantitative Feld,  $\vec{r}$  die Position des Partikels  $j$  und  $W$  eine Kernel-Funktion ist.

Die in den Navier-Stokes Gleichungen vorkommenden kontinuierlichen Anteile für die Druckverteilung und die Viskositätsverteilung können durch eine SPH berechnet werden. Dies führt nach Anwendung des Gradienten und des Laplace-Operators zu:

$$f_i^{\text{Druck}} = -\nabla p(r_i) = -\sum_j m_j \frac{p_j}{\rho_j} \nabla W(\vec{r}_i - \vec{r}_j, h) \quad (3.14)$$

und

$$f_i^{\text{Viskosität}} = \nu \Delta \mathbf{v}(\mathbf{r}_i) = \sum_j m_j \frac{v_j}{\rho_j} \Delta W(\vec{r}_i - \vec{r}_j, h) \quad (3.15)$$

Aus Gründen der Stabilität werden in [61] die nicht symmetrischen Anteile durch symmetrische ersetzt. Dies führt zu:

$$f_i^{\text{Druck}} = -\sum_j m_j \frac{p_i + p_j}{2\rho_j} \nabla W(\vec{r}_i - \vec{r}_j, h) \quad (3.16)$$

und

$$f_i^{\text{Viskosität}} = \sum_j m_j \frac{v_i + v_j}{\rho_j} \Delta W(\vec{r}_i - \vec{r}_j, h) \quad (3.17)$$

Mit SPH können keine inkompressiblen Flüssigkeiten simuliert werden. Abschließend muss erwähnt werden, dass die SPH Formulierung nicht unbedingt physikalisch korrekte Ergebnisse in Bezug auf die Symmetrie der Kräfte und Momenterhaltung liefert.

**Lennard-Jones-Potential** Eine weitere partikelbasierte Methode, Flüssigkeiten zu simulieren, stellt das Lennard-Jones-Potential (Gleichung 3.18) dar. Das Lennard-Jones-Potential stammt aus der physikalischen Chemie und beschreibt das Anziehungsverhalten von ungeladenen Teilchen in Gasen und Flüssigkeiten.

$$V(r) = -E_m \left( \frac{r_m^{12}}{r} - 2 \frac{r_m^6}{r} \right) \quad (3.18)$$

Dabei ist  $r$  der euklidische Abstand zwischen zwei Partikeln und  $V(r)$  liefert die Kraft zwischen diesen beiden Partikeln. Über  $E_m$  lässt sich die Energie im Minimum einstellen und über  $r_m$  den Abstand für den das Minimum angenommen wird.

Das Lennard-Jones-Potential weist folgende Eigenschaften auf: Es erzeugt für geringe Partikelabstände abstoßende, für mittlere Abstände anziehende Kräfte. Für sehr große Abstände strebt es gegen Null. Diese Eigenschaft kann ausgenutzt werden, um die Anzahl der zu berechnenden Kraftwerte pro Partikel zu minimieren, indem für jeden Partikel nur eine begrenzte Nachbarschaft berechnet wird. Da andererseits das Lennard-Jones-Potential für sehr geringe Abstände gegen unendlich strebt, ist es angebracht den Maximalwert zu beschränken, um die numerische Stabilität zu gewährleisten.

## 3.3 Numerische Integration

Die zeitliche numerische Integration ist die Grundlage jeder dynamischen Simulation. Sie ermöglicht aus den zum Zeitpunkt  $t$  aktuellen Partikelpositionen, -geschwindigkeiten und -beschleunigungen die Partikeleigenschaften für den nächsten Zeitpunkt  $t + \Delta t$  zu schätzen. Dabei wird  $\Delta t$  die Schrittweite genannt. Numerische Integrationsmethoden sind im Allgemeinen fehlerbehaftet. Dies erfordert eine sensible Wahl des Integrationsverfahrens und des Zeitschritts. Numerische Integrationsmethoden werden in explizite und implizite Methoden unterteilt.

**Explizite Methoden** Bei der Berechnung des nächsten Zustands berücksichtigen explizite Methode nur die bereits berechneten Werte. Dadurch neigen sie verstärkt zu Instabilität. Diese Fehler sind abhängig vom gewählten Integrationsverfahren sowie der Größe des Zeitschritts. Durch diesen Fehler wird dem simulierten System Energie hinzugefügt, wodurch es leicht zu Instabilitäten kommen kann.

Explizite Verfahren lassen sich elementweise anwenden. Sie benötigen für die zeitliche Integration keinerlei Information ihrer Nachbarschaft. Als anschauliches Beispiel soll hier die explizite Euler-Methode dienen.

Bei der Euler-Methode wird ausgehend vom 2. Newtonschen Gesetz (Gleichung 3.19) die Partikelposition  $\vec{x}_i(t + \Delta t)$  und -geschwindigkeit  $\vec{v}_i(t + \Delta t)$  zum Zeitpunkt  $t + \Delta t$  geschätzt:

$$\vec{a}_i(t) = \dot{\vec{v}}_i(t) = \ddot{\vec{x}}_i(t) = F_i(\vec{x}, \dot{\vec{x}}, t) \quad (3.19)$$

$$\vec{x}_i(t + \Delta t) = \vec{x}_i(t) + \vec{v}_i(t) * \Delta t \quad (3.20)$$

$$\vec{v}_i(t + \Delta t) = \vec{v}_i(t) + \dot{\vec{v}}_i(t) * \Delta t \quad (3.21)$$

Dabei ergibt sich ein Fehler von  $O(\Delta t^2)$ . Der Fehler lässt sich durch andere Methoden reduzieren, jedoch nicht vermeiden. Für eine stabile Simulation muss der Fehler gering gehalten werden. Dies gelingt bei expliziten Methoden nur für kleine  $\Delta t$ .

Einen Überblick über die in der VRM-Bibliothek implementierten Algorithmen gibt Grimm [32].

**Implizite Methoden** Bei der Berechnung des nächsten Zustands beziehen implizite Methoden den nächsten Zustand mit ein (Gleichung 3.24). Dadurch entsteht ein Gleichungssystem, das in jedem Simulationsschritt gelöst werden muss. Durch diese Einbeziehung des nächsten Zustands sind sie wesentlich fehlertoleranter, jedoch aufwendiger in der Berechnung.

$$\vec{a}_i(t + \Delta t) = \dot{\vec{v}}_i(t + \Delta t) = \ddot{\vec{x}}_i(t + \Delta t) = F_i(\vec{x}(t + \Delta t), \dot{\vec{x}}(t + \Delta t), t) \quad (3.22)$$

$$\vec{v}_i(t + \Delta t) = \vec{v}_i(t) + \dot{\vec{v}}_i(t + \Delta t) * \Delta t \quad (3.23)$$

$$\vec{x}_i(t + \Delta t) = \vec{x}_i(t) + \vec{v}_i(t + \Delta t) * \Delta t \quad (3.24)$$

$F_i(\vec{x}(t + \Delta t), \dot{\vec{x}}(t + \Delta t), t)$  ist nicht bekannt, da es Knotenpositionen zum Zeitpunkt  $t + \Delta t$  benötigt. Die Gleichungen sind somit nicht mehr unabhängig von den beteiligten Knoten und fließen mit in die Berechnung ein. Für Implementierungsdetails sei auf [9] verwiesen.

In der VRM-Bibliothek wurden noch keine impliziten Algorithmen implementiert.

## 3.4 Kollisionsbehandlung

Viele computergestützte Anwendungen benötigen eine Kollisionsbehandlung, einige davon sind aus dem Bereich computergestütztes Design, Robotik, Automation, Computergrafik und Simulation. Einen Überblick findet man in [52].

Simulierte Objekte besitzen von sich aus keine impliziten Kollisionseigenschaften. Diese müssen immer wieder neu berechnet werden. Jedes Objekt in der virtuellen Umgebung befindet sich zu einem bestimmte diskrete Zeitpunkt  $t$  an einem bestimmtem Ort  $\vec{x}(t)$ . Befinden sich zur selben Zeit  $t$  zwei oder mehr Objekte am selben Ort  $\vec{x}(t)$ , kommt es zu einer Kollision, die entsprechend aufgelöst werden muss. Man benötigt zwei Schritte: die *Kollisionserkennung* und die *Kollisionsantwort*. Unter Kollisionserkennung versteht man das Finden von Überschneidungen von geometrischen Körpern im simulierten Raum. Durch die Kollisionsantwort werden die gefunden Kollisionen aufgelöst, so dass es zu keinen Überschneidungen mehr kommt.

Die Wahl des Kollisionserkennungsalgorithmus hängt von der Modellrepräsentation in der virtuellen Realität ab.

Die Arbeit beschränkt sich auf Objekte, die eine polygonale Darstellung besitzen, d. h. die Objekte werden aus Knoten, Kanten und Oberflächen aufgebaut. Mit ihnen lassen sich sowohl starre wie deformierbare Objekte visualisieren.

### 3.4.1 Kollisionserkennung

Die Kollisionserkennung deckt ein weites Feld ab und wurde in vielen Arbeiten ausführlich untersucht. Einen umfangreichen Überblick bietet [96]. Die Anforderungen an die Kollisionserkennung sind von der speziellen Anwendung abhängig. Bei einer offline Bewegungsanalyse eines Roboterarmes in einer Produktionsstraße muss die Berechnungsqualität so gut sein, dass jegliche Kollision erkannt wird. Die Rechenzeit ist dabei zweitrangig, da die Berechnung meist nur einmal durchgeführt werden muss. Bei Echtzeit Operationssimulationen ist die Rechenzeit der Hauptaspekt. Die Wahl der passenden Berechnungsverfahren hängt zusätzlich von den verwendeten Objekten ab. Für starre Objekte eignen sich andere Methoden wie für deformierbare Objekte.

Die Berechnung der Kollisionserkennung ist ein aufwendiger Teil der gesamten Simulation, deswegen soll hier kurz auf die Probleme und der möglichen Lösungsansätze eingegangen werden.

**All-Pairs-Problem** Der Rechenaufwand für die Kollisionserkennung ergibt sich aus der Anzahl, der in der virtuellen Umgebung verwendeten Objekte. Wenn jedes Objekt auf eine Kollision mit allen anderen Objekten getestet werden muss, gilt für  $n$  bewegliche Objekte, dass  $\binom{n}{2}$  Kollisionen geprüft werden müssen. Somit wäre eine Brute-Force-Methode schon in Szenen mit wenigen Objekten kaum durchführbar. Nach Bandi [5] ist die Anzahl der wirklich stattfinden Kollisionen pro Zeitschritt meist wesentlich geringer als die maximale mögliche Anzahl.

Die eigentliche Herausforderung der Kollisionserkennung besteht somit darin, Algorithmen zu entwickeln, die Kollisionen in einer angemessenen Zeitspanne finden. Da, wie schon erwähnt, in Bezug auf die zu untersuchenden Objekte die Anzahl der Kollisionen pro Zeitschritt gering ist, kann der Suchraum durch geeignete Voruntersuchungen wesentlich verkleinert werden. Ein weit verbreiteter Ansatz ist die Unterteilung in zwei Phasen. Man bezeichnet diese zwei Phasen als ein Weit-Phase (engl. *Broad-Phase*) und als Nah-Phase (engl. *Narrow-Phase*).

#### 3.4.1.1 Broad-Phase

In der Broad-Phase werden zunächst durch einfache Tests die Objekte oder die Objektteile bestimmt, die möglicherweise kollidieren. Die in der Broad-Phase als mögliche Kollisionen erkannten Teile werden dann in der Narrow-Phase weiter untersucht, um eine Kollision definitiv zu erkennen oder auszuschließen.

Die Broad-Phase lässt sich in zwei Gruppen einteilen, in Raum- und Objekt-basierte.

#### Spatial Subdivision (raumbasiert)

Bei der Spatial Subdivision wird die simulierte Szene in kleinere räumliche Einheiten aufgeteilt. Die Objekte werden nach jedem Zeitschritt in diese Einheiten einsortiert. Belegen dabei mehrere Objekte dieselbe räumliche Einheit, ist eine Kollision möglich. Diese so gefundenen interessanten Teile werden weiter untersucht. Es wird zwischen regelmäßigen und unregelmäßigen Gittern unterschieden.



Wird der zu untersuchende Raum in Einheiten gleicher Größe eingeteilt, erhält man ein regelmäßiges Gitter (engl. *Voxel-Grid*). Bevorzugt werden rechtwinklige Strukturen eingesetzt. Das Einsortieren kann in linearer Zeit erfolgen, wenn nur Knoten in das Gitter einsortiert werden. Die Auswertung ist abhängig von der Zellengröße. Bei großen Zellen kann die Anzahl der zu untersuchenden Zellen stark zunehmen. Markiert man außerdem noch die Zellen als besetzt, die durch die Flächen des Objektes belegt werden, übt dies einen weiteren Einfluss auf die zum Einsortieren benötigte Zeit aus. Das Ermitteln der geeigneten Zellengröße ist, vor allem bei stark schwankender Objektgröße, schwierig. Ein weiterer Nachteil ist, dass diese Datenstruktur, wenn sie zu Beginn angelegt wird, sehr viel Speicherplatz benötigt. Teschner et al. [95] schlagen ebenso wie Overmars [70] deshalb ein Hashed-Verfahren zur Speicherung vor. Durch hierarchische Strukturen lässt sich das Problem der statisch gewählten Zellengröße bewältigen. In [5] wird eine Octtree-Struktur dafür vorgeschlagen. Die Zellen werden solange weiter unterteilt, bis keine Kollision mehr festgestellt wird oder eine vorgegebene minimale Zellengröße erreicht wird. Wenn in dieser Auflösungsebene immer noch Zellen von mehreren Objekten belegt sind, werden diese genauer auf Kollisionen untersucht.

Existiert keine vorgegebene Struktur, in die die Elemente einsortiert werden, spricht man von unregelmäßigen Gittern. Die Struktur ergibt sich hierbei durch die räumliche Anordnung der Elemente. Beispiele sind BSP-Bäume (engl. *Binary Space Partitioning*) und verallgemeinerte k-d-Bäume. Bei BSP-Bäumen wird der Raum in eine Reihe von Halbräumen zerlegt, die die Hierarchie des Baumes erzeugen. In [55] wird ein Verfahren vorgeschlagen, bei dem der BSP-Baum dynamisch an die Objektgröße angepasst wird, so dass keine Kopien des BSP-Baumes für alle vorkommenden Objektgrößen mehr angelegt werden müssen.

### **Bounding Volume Hierarchy (objektbasiert)**

Eine andere Herangehensweise die Kollisionserkennung zu beschleunigen, bietet der objektbasierte Ansatz. Hierbei wird die Objektgeometrie der in der Szene vorkommenden Objekte mit geeigneten konvexen Hüll-Objekten (engl. *bounding objects*) abstrahiert. Da die Hüll-Objekte das gesamte Objekt beinhalten, kann es nur zu einer Kollision kommen, wenn die Hüll-Objekte kollidieren. Dies eignet sich vor allem für nicht deformierbare Objekte, bei denen sich die Objektgeometrie nicht mehr ändert. Dann können die notwendigen Berechnungen in einem Vorverarbeitungsschritt ausgeführt werden. Der objektbasierte Ansatz ist somit gut geeignet, um Operationsinstrumente zu approximieren. Als Hüll-Objekte werden einfache geometrische Objekte wie Kugeln, Zylinder, Quader, DOPs und konvexe Hüllen verwendet. Diese ermöglichen eine schnelle Überprüfung auf Kollisionen. Im Allgemeinen exis-

tiert kein *optimales* Hüll-Objekt. Die Hüll-Objekte müssen immer an die vorliegende Szene angepasst werden, somit gibt es für jede spezielle Szene ein optimales Hüll-Objekt.

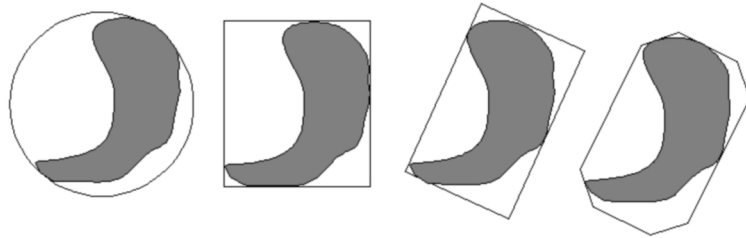


Abbildung 3.1: Für die gezeigten Hüll-Objekte (von links nach rechts: Kugel - AABB - OBB - k-DOP) nimmt die Approximationsgüte und der Rechenaufwand zu.

Jedoch gilt der in Abbildung 3.1 dargestellte Zusammenhang zwischen Approximationsgüte und Rechenzeit. Durch die Anordnung in hierarchischen Baumstrukturen lässt sich die Approximationsgüte erhöhen und die Rechenzeit für die Kollisionserkennung gering halten. Durch die Baumstruktur lässt sich schnell der Teil des Objektes bestimmen, der von der möglichen Kollision betroffen ist. In der Vergangenheit wurden schon viele Hüll-Objekte Hierarchien (engl. *bounding volume hierarchy* - BVH ) verwendet und untersucht, dazu gehören Kugeln [39], Quader (OBB [29], AABB [47], [98]) und k-DOPs [46]. Eine schnelle Kollisionsberechnung ermöglichen Kugeln, da nur der Mittelpunkt  $\vec{m}$  und der Radius  $r$  berücksichtigt werden muss. Jedoch lässt sich nicht jedes Objekt optimal durch sie approximieren und das automatische Erstellen einer BVH ist zeitintensiv [39]. Mit Quadern lassen sich zwei unterschiedliche Arten Hüll-Objekte erstellen, achsenparallele, sogenannte AABB (engl. *axis-aligned bounding box*) oder objektorientierte, auch OBB (engl. *oriented bounding box*) genannt. Achsenparallele Quader sind an den Koordinatenachsen des Weltkoordinatensystems ausgerichtet, dies ermöglicht schnelle Überlappungstests unter der Inkaufnahme der schlechteren Objektapproximation. Ebenso müssen die Ausmaße des Quaders in jedem Schritt neu berechnet werden. Im Gegensatz dazu werden die OBBs an den Hauptachsen des zu approximierenden Objektes orientiert. Ein OBB kann in  $O(n^3)$  Rechenschritten an ein Objekt mit  $n$  Knoten angepasst werden. Gottschalk et al. gehen in [29] ausführlich auf den Aufbau von OBB-Bäumen ein. Um die OBBs an das Objekt anzupassen, verwenden sie neben den Eigenvektoren der Kovarianz-Matrix zusätzlich die konvexe Hülle. Dadurch können Ausprä-

gungen des Objektes, die sich negativ auf die Passform des OBB auswirken, unterdrückt werden. Um den OBB-Baum aufzubauen, verwenden sie einen rekursiven Algorithmus und geben eine Rechenzeit von  $O(n \log^2 n)$  für  $n$  Dreiecke an, wenn sie die konvexe Hülle berücksichtigen und  $O(n \log n)$  ohne die konvexe Hülle. Bei starren Objekten lassen sich die OBBs mit dem Objekt bewegen und müssen nicht neu erstellt werden.

Noch genauer als mit einer OBB lässt sich ein Objekt mit einem  $k$ -DOP approximieren. Dabei wird das Objekt durch  $k$  Halbräume, wobei  $k$  eine gerade Anzahl ist, eingegrenzt. In [46] wird eine  $k$ -DOP Hierarchie aufgebaut, die für jedes Objekt dieselben  $k$ -Richtungen benutzt. Ein Objekt ist somit vollständig durch  $k/2$  Intervalle beschrieben. Falls sich alle Intervalle überlappen, liegt eine Kollision vor und es wird die nächste Hierarchiestufe getestet, es sei denn die unterste Ebene wurde erreicht. Dort erfolgen Primitivtests. Die Untersuchungen in [46] haben gezeigt, dass  $k$ -DOPs durchaus für BVHs geeignet sind und gängigen OBB Implementierungen ebenbürtig sind.

Klosowski et al. [46] geben eine allgemeine Kostenfunktion für die Verwendung von BVHs an. Diese bezieht neben den Kollisionstests auch die Zeit für den Aufbau der Hierarchie mit ein:

$$T = N_v \times C_v + N_p \times C_p + N_u \times C_u \quad (3.25)$$

Dabei gibt  $T$  die Gesamtkosten an,  $N_v$  die Anzahl der zu testenden Hüll-Objekt-Paare,  $C_v$  die Kosten für einen paarweisen Kollisionstest,  $N_p$  die Anzahl der zu testenden Primitiv-Paare,  $C_p$  die Kosten für einen paarweisen Primitivtest,  $N_u$  die Anzahl der zu aktualisierenden Hierarchieanteile und  $C_u$  die Kosten für die Aktualisierung. Hier wird deutlich, dass BVHs nur bedingt für deformierbare Objekte geeignet sind, da die Kosten für das Aktualisieren der BVH die Dauer der Kollisionsberechnung mitunter stark beeinflussen.

#### 3.4.1.2 Narrow-Phase

Die Nah-Phase (engl. *Narrow-Phase*) behandelt die genaue Bestimmung der Kollisionen, bei polygonalen Modellen wird dazu eine Reihe von *Primitivtests* durchgeführt. Dies sind Tests von z. B. Knoten-Knoten-, Knoten-Fläche-, Kante-Kante- oder Kante-Knoten-Kollisionen. Es steht ausreichend Literatur<sup>3</sup>, die sich mit der effizienten Berechnung dieser Primitivtests beschäftigt, zur Verfügung.

Neben den Primitivtests gibt es eine Reihe von Algorithmen, die Kollisionen zwischen Polyedern finden. Erste Algorithmen [50] verwendeten Lineare Programmierung, um den Abstand zwischen Polyedern zu bestimmen. Jedoch

---

<sup>3</sup>Es sei verwiesen auf Ericson [22].

sind LP-Solver sehr aufwendig in der Berechnung einer trennenden Ebene. Dies führte zu Algorithmen, die dies zu vermeiden versuchen. Der Simplex-basierte GJK Algorithmus, benannt nach seinen Entwicklern [27], sucht einen Simplex, der durch die Vertices der Minkowski Difference der beteiligten Polyeder gegeben ist und den Ursprung beinhaltet oder nahe dem Ursprung liegt. Ist der Ursprung nicht beinhaltet, so entspricht der Abstand zum Ursprung dem Abstand der Polyeder. Ist der Ursprung beinhaltet, durchdringen sich die Polyeder.

Ein weiterer Simplex-basierter Algorithmus ist der Lin-Canny Algorithmus [51], der über die Oberfläche des erstellten Simplex läuft und Kollisionen bei einem Mindestabstand  $\epsilon$  erkennt. Durchdringungen werden nicht behandelt.

Der V-Clip [58] Algorithmus, der als Nachfolger des Lin-Canny Algorithmus angesehen werden kann, soll die Unzulänglichkeiten beheben. V-Clip kann Durchdringungen feststellen, ist numerisch robust und einfach zu implementieren.

Als weitere schnelle Kollisionserkennungsbibliothek kann I-Collide [17] aufgeführt werden. I-Collide ist auf die gleichzeitige Kollisionserkennung bei  $n$  konvexen Polyedern ausgelegt. Anstatt  $O(n^2)$  Vergleichen benötigt der Algorithmus  $O(n + m)$ , wobei  $m$  die Anzahl sehr naher Polyeder ist.

#### 3.4.1.3 Weitere Methoden

Auf dem Gebiet der Kollisionserkennung wird weiterhin viel Forschungsarbeit betrieben und es werden immer wieder neue Methoden entwickelt. Einige davon sollen hier erwähnt werden.

**Distance field collision detection** Die Ursprünge der Distanzfelder liegen bei der 3D Daten Visualisierung (Voxelrendering) und den Level-Set Methoden. Dabei wird eine räumliche Repräsentation erstellt, durch die es möglich ist, das Innere und Äußere eines Objektes zu unterscheiden. Über eine skalare Funktion:

$$D : R^3 \rightarrow R \quad (3.26)$$

wird die Oberfläche als Null-Ebenen-Set definiert, dass gilt:

$$D(p) = \text{sign}(p) * \text{dist}(p) \quad (3.27)$$

wobei  $\text{sign}(p)$  negativ ist, wenn  $p$  innerhalb des Objektes liegt und  $\text{dist}(p)$  den Abstand zur Oberfläche angibt. Dafür müssen geeignete Methoden genutzt werden, die die räumliche Repräsentation der Objekte möglichst gut

abbilden. Bei sich zeitlich verändernden Objekten dürfen die Kosten für die Neuerstellungen nicht zu hoch sein. Beispiele für die Generierung solcher räumlichen Repräsentationen gibt es in [38] und [24].

**Stochastic collision detection** Ein anderer Ansatz den Suchraum zu minimieren wird durch die stochastische Kollisionserkennung erreicht. Über stochastische Methoden wird dabei eine Vorauswahl der zu untersuchenden Kollisionsobjekte getroffen. Da es sich hierbei um ein *unexaktes* Verfahren handelt, ist es nicht ausgeschlossen, dass Kollisionen unerkannt bleiben. Barzel [10] hat gezeigt, dass es bei vielen Echtzeitanwendungen nicht auf physikalisch-korrekte, sondern auf physikalisch-plausible Kollisionsantworten ankommt, vor allem, wenn dadurch die Randbedingung der Echtzeit nicht verletzt wird. Es kann somit für manche Szenarien von Vorteil sein, solche stochastische Methoden einzusetzen.

Als Beispiel kann die Methode von Lin [51] genannt werden. Der Ansatz nutzt aus, dass bei einer Kollision meist nur wenige Punkte direkt miteinander kollidieren. Es fließen deshalb nicht alle Punkte in die Berechnung ein. Es werden zufällige Paare von Punkte gewählt, deren Abstand berechnet wird. Unterschreitet der Abstand eines Punktpaares einen vorgegeben Mindestabstand, wird dieses Punktpaar als *aktive Verbindung* gespeichert. Im nächsten Durchgang werden wieder zufällige Paare gewählt und der Abstand berechnet. Ebenso wie aktive Verbindungen erstellt werden, können diese wieder gelöst werden, falls der Abstand größer wird. Die aktiven Verbindungen werden dann genauer auf Kollisionen untersucht.

**GPU für Kollisionserkennung** Eine weitere Entwicklung ist der Einsatz von Grafikkarten zur Berechnung der Kollisionserkennung [30], [49] und [31]. Dabei macht man sich die hohe Rechenleistung sowie die Parallelität der aktuellen Grafikkarten zunutze. Images-Space Ansätze [102] sind eine Möglichkeit Kollisionen mit Grafik-Hardware zu berechnen, meist wird dazu der *z-Puffer* oder der *stencil-Puffer* verwendet. Durch die neuen Programmier-Interfaces wie CUDA (engl. *Compute Unified Device Architecture*) von Nvidia oder dem kommenden OpenCL (engl. *Open Computing Language*) ist Software-Entwicklung für Grafikkarten einfacher geworden. Dies erleichtert die Implementierung für die GPGPU (engl. *General-Purpose computation on Graphics Processing Units*). Für eine erfolgreiche Portierung von Standard-Algorithmen muss die Parallelität ausgenutzt werden, denn die parallele Ausführung kann einen Algorithmus stark beschleunigen.

**Continuous Collision Detection** Dieser Ansatz dient nicht der Beschleunigung der Kollisionserkennung, sondern der Verbesserung der Kollisionsgüte.

Alle bisher besprochenen Methoden behandeln Kollisionen als diskrete Ereignisse zu bestimmten Zeitpunkten  $t$ . Nach einem Simulationsschritt wird dann wieder zum Zeitpunkt  $t + h$  auf neue Kollisionen geprüft. Dies stellt jedoch nicht sicher, dass es in der Zeitspanne  $h$  zu keiner Kollision kam, dies ist auch als Tunneleffekt bekannt.

Bei der Continuous Collision Detection werden die Positionen der zu behandelnden Objekte zwischen  $t$  und  $t + h$  interpoliert und auf Kollisionen geprüft. Bewegt sich ein nicht deformierbares Objekt durch den Raum, überschreitet es von einem zum nächsten Zeitschritt ein bestimmtes Volumen. Wenn sich für zwei Objekte diese Volume nicht überlappen, ist sichergestellt, dass keine Kollision vorlag. Redon et al. verwenden diese Methode in [73] für OBBs.

#### 3.4.2 Kollisionsantwort

Die Kollisionsantwort berechnet das Verhalten der beteiligten Objekte bei einer erkannten Kollision. Sie muss dafür Sorge tragen, dass es zu einer Auflösung der Kollision kommt, so dass das Gesamtsystem wieder einen überschneidungsfreien Zustand annimmt.

Bei der Kollisionsantwort muss zwischen Methoden für Festkörper und deformierbare Objekte unterschieden werden. Es existiert eine Reihe von Ansätzen zur Kollisionsantwort. Die drei verbreitetsten Methoden für Festkörper sollen hier kurz eingeführt werden. Es sind dies die Penalty-, die Constraint- und die Impuls-Methode. Für deformierbare Objekte sind zwei Ansätze weit verbreitet: die Penalty-Methode und die verschiebungsorientierten Ansätze.

**Klassifizierung der Kollisionen** Im Folgenden wird zwischen *kollidierenden*, *ruhenden* und *trennenden* Kollisionen unterschieden. Über die relativen Geschwindigkeiten der Objekte lässt sich ermitteln, ob es sich um kollidierende, ruhende oder trennende Kontakte handelt. Für die relative Geschwindigkeit  $v_c$  in einem Kontaktpunkt  $P$  gilt:

$$v_c = \vec{n}_b(\dot{\vec{r}}_a - \dot{\vec{r}}_b) \quad (3.28)$$

Wobei  $\dot{\vec{r}}_a$  die Geschwindigkeit des Kontaktpunktes  $\vec{r}_a$  in lokalen Koordinatensystems des Objektes  $a$  ist,  $\dot{\vec{r}}_b$  entsprechend die des Objektes  $b$  mit der Normalenrichtung  $\vec{n}_b$  der Oberfläche des Objektes  $b$  am Kontaktpunkt.

Bei der Kollision handelt es sich dann um eine

- **kollidierende Kollision**, wenn die relative Kontaktgeschwindigkeit  $v_c < 0$  ist. Dies erfordert eine sofortige Änderung der Geschwindigkeit, damit sich die Objekte nicht weiter durchdringen.
- **ruhende Kollision**, wenn die relative Kontaktgeschwindigkeit  $v_c = 0$  ist. Die Körper befinden sich relativ zu einander in Ruhe, da weiterhin äußere Kräfte, wie z. B. die Erdbeschleunigung, auf die Körper wirken, muss eine ausgleichende Kraft berechnet werden, so dass die Objekte in diesem Zustand verharren, ohne sich zu durchdringen oder abzustößen.
- **trennende Kollision**, wenn die relative Kontaktgeschwindigkeit  $v_c > 0$  ist. In diesem Fall muss keine Kollisionsbehandlung durchgeführt werden, da sich die Objekte von einander weg bewegen und es zu keiner Durchdringung kommt.

In den letzten Jahrzehnten wurden verschiedene Verfahren zur Kollisionsauflösung entwickelt. Es sollen hier die gebräuchlichsten kurz vorgestellt werden.

#### 3.4.2.1 Penalty-Methode

Bei Penalty basierten Ansätzen werden erkannte Kollisionen sanktioniert. Diese Sanktionen werden durch Penalty-Kräfte realisiert. Die Penalty-Kraft wirkt in Richtung der Oberfläche, die nicht durchdrungen werden darf. Solch eine Penalty-Kraft lässt sich durch eine steife lineare Feder-Kraft  $k_p$  modellieren. Diese Feder wird temporär an der Kontaktstelle eingefügt und zieht das kollidierende Objekt wieder an die Oberfläche des anderen Objektes. Dies ist ein sehr anschauliches Verfahren. Schwierig ist die Wahl des passenden Wertes für  $k_p$ , der im Allgemeinen sehr hoch sein sollte, um eine gewisse Oberflächensteifigkeit zu erreichen. Dies führt zu steifen Differentialgleichungen, die numerisch instabil werden. Nur mit kleinen Zeitschritten lässt sich das System stabil halten [6]. Wählt man  $k_p$  zu gering, können sich die Objekte zu leicht durchdringen. Die durch die Penalty-Methode ermittelten Kräfte sind keineswegs exakt, aber sie bietet eine schnell zu berechnende Abschätzung der Kollisionskräfte.

Penalty-Ansätze wurden schon früh in der Computersimulation verwendet. Terzopoulos et al. [91] schlagen eine eindringtiefenabhängige Penalty-Kraft vor.

In [64] wird von Moore und Wilhelms vorgeschlagen, Federn mit einer Federkonstante  $k$  zur Berechnung der Kollisionsantwort zu benutzen. Des Weiteren lässt sich  $k$  über einen zusätzlichen Wert  $\epsilon = [0, 1]$  für annähernde bzw. entfernende Objekte einstellen:

$$k_{\text{entfernend}} = \epsilon k_{\text{annähernd}} \quad (3.29)$$

Für  $\epsilon = 0$  handelt es sich um eine vollständig unelastische und für  $\epsilon = 1$  um eine vollständig elastische Kollision.

Ein neueres Verfahren, vorgestellt in [36], berücksichtigt nicht die Eindringtiefe, sondern das gesamte sich überlappenden Volumen zur Berechnung der Penalty-Kräfte.

Heidelberger et al. [37] benutzen ebenfalls einen linearen Penalty-Ansatz für die Auflösung von Kollisionen. Dabei verwenden sie nur Kollisionspunkte, die direkt mit mindestens einem Nicht-Kollisionspunkt verbunden sind. Nur diese fließen in die Berechnung der Kräfte ein.

#### 3.4.2.2 Constraint-Methode

Im Gegensatz zu Penalty-Methoden werden bei dieser Methode Zwangskräfte exakt berechnet, so dass es für ruhende Kontakte zu keinen Durchdringungen kommt.

Über Zwangsbedingungen lässt sich das physikalische Verhalten von Objekten im Kollisionsfall beschreiben. Für einen gemeinsamen Kontaktpunkt  $P$  gilt für den Abstand  $d$ :

$$d = \vec{n}_b(\vec{r}_a - \vec{r}_b) \quad (3.30)$$

mit der Normalenrichtung  $\vec{n}_b$ , der Kontaktoberfläche und den beiden Kontaktpunkten  $\vec{r}_a$  und  $\vec{r}_b$  im lokalen Koordinatensystems des entsprechenden Objektes. Damit es zu keinen Durchdringungen kommt, müssen folgende Bedingungen erfüllt sein:

- die Beschleunigung  $\ddot{d} \geq 0$  am Kontaktpunkt  $P$ , um Durchdringung zu verhindern,
- der Betrag der Kontaktkräfte in Normalenrichtung  $f \geq 0$ , um Haftung zu vermeiden und
- es darf keine Kraft mehr wirken, wenn gilt  $\ddot{d} = 0$  und  $f \geq 0$  oder  $\ddot{d} > 0$  und  $f = 0$  anders ausgedrückt  $f * \ddot{d} = 0$ .

Dies sind die Randbedingungen für einen Kontaktpunkt. Bei mehreren Kontakten zwischen mehreren Objekten beeinflussen sich die Kontaktkräfte gegenseitig. Stellt man für jeden Kontaktpunkt eine Gleichung auf, ergibt sich ein Gleichungssystem, dessen Lösung die exakten Kontaktkräfte für alle Kontaktpunkte liefert.



In Matrixschreibweise ergibt sich für  $n$  Kontaktpunkte:

$$\mathbf{A}\mathbf{f} + \mathbf{b} = \ddot{\mathbf{d}} \quad \text{für } \mathbf{f}, \mathbf{d} \geq 0 \quad \text{und } \mathbf{f} * \mathbf{d} = 0 \quad (3.31)$$

Die Matrix  $\mathbf{A} \in \mathbb{R}^{n \times n}$  ist symmetrisch und positive semidefinit, sie beinhaltet Informationen über die Kontaktpunkte und die Massen der beteiligten Objekte. Im Vektor  $\mathbf{b} \in \mathbb{R}^n$  werden die externen Kräfte und Momente gespeichert. Aus der Mathematik ist solch ein Gleichungssystem als LCP (engl. *Linear Complementarity Problem*) bekannt. Da die Matrix  $\mathbf{A}$  positiv semidefinit ist, ist das LCP lösbar.

Für die Lösung kann auf verschiedene Algorithmen zurückgegriffen werden. Eine direkte Lösung bietet der Lemke Algorithmus ( $O(n^4)$ ), iterative Lösungen der Projected Gauss Seidel ( $O(n^2)$ ), der Conjugate Projected Gradient ( $O(n^2)$ ) und der Smooth Damped Newton Algorithmus.

Meist wird das Verfahren nur für ruhende Kollisionen eingesetzt, in [6] wird eine Erweiterung vorgestellt, die es ermöglicht, das Verfahren für kollidierende Kollisionen zu verwenden. Das angepasste System hat folgende Form:

$$\mathbf{A}\mathbf{j} + \mathbf{b} = \dot{\mathbf{d}} \quad \text{für } \mathbf{j}, \mathbf{d} \geq 0 \quad \text{und } \mathbf{j} * \mathbf{d} = 0 \quad (3.32)$$

wobei  $\mathbf{j}$  die zu berechnenden Impulse bezeichnet. In die Matrix  $\mathbf{A}$  und den Vektor  $\mathbf{b}$  fließen dann die Objektgeometrien und die Geschwindigkeiten ein.

In [7] wird ein Ansatz vorgestellt, bei dem das Modell um Reibung erweitert wird. Dabei wird gezeigt, dass die Matrix  $\mathbf{A}$  für statische Reibung weiterhin positiv semidefinit bleibt und es somit immer eine Lösung für das System gibt. Fügt man dynamische Reibung hinzu, ist die Matrix  $\mathbf{A}$  nicht mehr unbedingt positiv semidefinit. Da die Richtung der Reibungskraft immer der Bewegungsrichtung entgegen wirkt, kann die Matrix negative Anteile beinhalten und verliert ihre Symmetrie. Somit kann es sein, dass das System unlösbar wird. Deshalb schlägt Baraff [7] den Lemke Algorithmus zum Lösen des LCP vor, da dieser auch unter den gegebenen Bedingungen terminiert. Des Weiteren gibt er an, dass diese Betrachtungen eher theoretischer Natur sind, da sie nur für unnatürlich große Reibungskoeffizienten auftreten.

### 3.4.2.3 Impuls-Methode

Impulsbasierte Ansätze gehen auf die Arbeiten Keller [45] und Hahn [34] zurück. Eine ausführliche Beschreibung findet sich in der Dissertation von Mirtich [57].

Der Impuls  $\vec{p} = m\vec{v}$  stellt einen Zusammenhang zwischen der Kraft auf einen Körper und der Zeit, die benötigt wird diese zu übertragen, her:

$$\frac{d\vec{p}}{dt} = \frac{d(m\vec{v})}{dt} = \vec{f} \quad (3.33)$$

Kommt es zwischen zwei Festkörper zum Zeitpunkt  $t_c$  zu einer kollidierenden Kollision, so kommt es bei keiner abrupten Änderung der Relativgeschwindigkeiten zu einer Überschneidung der Objekte. Dies kann durch die Anwendung eines Impulses  $\vec{p}$  auf die Objekte verhindert werden. Der Impuls  $\vec{p}$  auf Festkörper der Masse  $m$  führt zu folgender Geschwindigkeitsänderung:

$$\Delta\vec{v} = \frac{\vec{p}}{m} \quad (3.34)$$

Beim impulsbasierten Ansatz werden sämtliche Kollisionen und Kontakte durch eine Reihe von Impulsen aufgelöst. Dabei werden Kollisionen als punktuelle Ereignisse aufgefasst, die durch einen entsprechend großen Impuls aufgelöst werden können. Wird eine Kollision zwischen einem Objektpaar erkannt, wird ein Impuls berechnet. Dieser Impuls  $\vec{p}$  muss eine Durchdringung verhindern und den physikalischen Gesetzen gehorchen. Der Impuls  $\vec{p}$  wird in entgegengesetzte Richtung zur aktuellen Geschwindigkeiten auf die Objekte angewendet. Mirtich berücksichtigt in seinem Ansatz den Energieverlust durch Deformation über die Hypothese von Poission:

$$p_{\text{total}} = (1 + e)p_{\text{mc}} \quad (3.35)$$

wobei  $p_{\text{total}}$  der Betrag des Impulses des Gesamtsystems ist und  $p_{\text{mc}}$  der Betrag der Impulse der Kollision. Für  $e = 1$  handelt es sich um einen vollkommen elastischen Stoß bei dem keine Energie verloren geht. Für  $e = 0$  wird die gesamte Energie absorbiert und die Objekte trennen sich nach der Kollision nicht. Ebenso modelliert er die Reibung durch das Coulomb'sche Reibungsgesetz.

Mirtich sieht den Vorteil dieser Methode darin, dass es nur einen Kollisionstyp gibt. Somit müssen keine unterschiedlichen Berechnungen durchgeführt werden. Er eignet sich selbst für ruhende Kontakte. Dabei erfährt ein ruhendes Objekt viele kleine Impulse, so dass es gerade zu keiner Durchdringung kommt. Diese werden auch „microcollisions“ genannt [60]. Problematisch erscheint ihm die Behandlung von anderen Verbindungen, wie z. B. Gelenken. Deshalb schlägt er in [56] eine Kombination aus Constraint- und Impuls-Methode vor. Dabei wird der Impuls durch die relative Objektgeschwindigkeit ermittelt. Die Auswirkungen des Impulses auf die Constraint-Objekte sind unklar und werden vernachlässigt.

In [75] wird eine Kollisionsbehandlung auf Basis der Impuls-Methode durchgeführt. Dabei werden nur paarweise Kollisionen berücksichtigt. Um neu entstandene Kollisionen aufzulösen, wird eine mehrstufige Kollisionsbehandlung

durchgeführt. Da in der Veröffentlichung aus dem Jahre 2003 keine genauen Angaben zur verwendeten Hardware gemacht wurden, kann nur davon ausgegangen werden, dass sie die zu diesem Zeitpunkt übliche Standard-Hardware benutzten. Als Rechenzeiten für 5 Kollisionsdurchgänge geben sie bei 500 (1000) Ringen, die auf ein Stabgitter fallen, 3 (7) Minuten pro Frame an.

#### 3.4.2.4 Verschiebungsorientierte Ansätze

Eine weitere Möglichkeit Kollisionen aufzulösen sind verschiebungsorientierte Ansätze. Dabei werden die Kollisionspunkte in einem Post-Simulationsschritt aus dem Objekt herausbewegt. Dies ist ein sehr spezieller Ansatz und führt nicht immer zum Erfolg, weil es leicht zu neuen Kollisionspunkten kommen kann, die sonst vielleicht nie entstanden wären.

Verschiebungsorientierte Ansätze kommen häufig bei deformierbaren Objekten zum Einsatz. Die lokalen Änderungen der kollidierenden Knoten wirken sich nicht direkt auf die globale Position des Objektes aus. Dadurch entstehen während der Kollisionsauflösung keine weiteren Kollisionen. So bestimmen Spillmann et al. [87] eine Kontaktoberfläche anhand der Kollisionstiefe. Neben den kollidierenden Knoten beziehen sie auch die Knoten der beteiligten Dreiecke mit ein, um eine kontinuierlichere Kollisionsbehandlung zu erzielen. In einem Vergleich mit einem Penalty basierten Ansatz konnten sie zeigen, dass ihr Ansatz wesentlich besser mit Objekten hoher Masse zurecht kommt und zu keinen weiteren Überschneidungen führt.

## 3.5 Modellierung topologischer Veränderungen

Topologische Veränderungen werden benötigt, wenn das zu simulierende Objekt während der Simulation geschnitten, gerissen oder zerbrochen wird. In den letzten Jahren hat die Anzahl dieser Ansätze stark zugenommen, so wurden diverse Reiß- und Schneide-Algorithmen für flächige und volumetrische Objekte entwickelt. Eine Einführung bieten Bruyns et al. [14]. Eine ausführliche Behandlung neuerer Ansätze bietet Steinemann in seiner Dissertation [89].

Grimm [32] gibt einen genauen Überblick, welche Methoden in Eyesi verwendet werden. Die von Grimm [32] vorgeschlagenen Methoden für Membranen wurden von ihm für den Einsatz für volumetrische Objekte erweitert. Diese Erweiterungen wurden noch nicht veröffentlicht. Sie werden jedoch für das später erwähnte Spalten der Linse eingesetzt.



# Kapitel 4

## Augenchirurgische Simulatoren

Mit der Entwicklung computergestützter Simulatoren für die Kataraktoperation beschäftigen sich weltweit einige Gruppen. Die einzelnen Projekte werden in diesem Kapitel vorgestellt. Neben diesen Arbeiten wird auch der Augenoperationssimulator Eyesi vorgestellt.

### 4.1 PhacoVision



Abbildung 4.1: Der PhacoVision<sup>®</sup> Simulator von Melerit. (Quelle: Melerit Medical)

Der Aufbau ist in Abbildung 4.1 zu sehen und wird in [48] und [81] beschrieben. Entwickelt wurde PhacoVision<sup>®</sup> von Melerit. Der Simulator besteht aus einem Personal Computer, Simulationssoftware und den Hardware-Interfaces zur Ein- und Ausgabe.

Die Simulation der Phakoemulsifikation ist in M-Base<sup>®</sup> geschrieben. Bei M-Base<sup>®</sup> handelt es sich um eine Simulationssoftware von Melerit AB die auf Cosmo 3d/Optimizer von Silicon Graphics aufsetzt.

Als virtuelle Instrumente stehen ein Phakoemulsifikationshandstück und ein Kernmanipulator zur Verfügung. Mit einem haptischen Eingabegerät werden für die beiden Instrumenten jeweils vier Freiheitsgrade gemessen (drei für die Translation und einer für die Rotation). Mit dem Mikroskopfußpedal lassen sich die x- und y-Ausrichtung sowie Zoom und Fokus beeinflussen. Ein Phakoemulsifikationsfußpedal dient zur Steuerung der Irrigation, Aspiration und des Ultraschalls. Über ein Ausgabe-Interface, das aus zwei OLEDs (engl. *Organic Light-Emitting Diode*) besteht, wird ein dreidimensionaler Eindruck der simulierten Szene vermittelt.

Der Simulator wurde in mehreren klinischen Studien getestet [82], [83], [84], [86] und [85]. Mit PhacoVision<sup>®</sup> ist es nicht möglich, die Phako-Prozedur als Ganzes durchzuführen. Sie wurde in drei Phasen aufgeteilt, die unabhängig bearbeitet werden können. Die einzelnen Phasen sind nach den durchzuführenden Arbeitsschritten bezeichnet. Es gibt eine *Sculpting*-, eine *Cracking*- und eine *Evakuierungsphase*. Beim Sculpting kann mit dem Phakoinstrument mittels Ultraschall ein Graben durch die Linse gezogen werden. Beim Cracking wird die Linse unter der Zuhilfenahme des Kernmanipulators gespalten. In der Evakuierungsphase entfernt der Lernende das Linsengewebe mit dem Phakoinstrument aus dem Auge.

Der Simulator wird derzeit von Melerit als kommerzielles Produkt vertrieben.

## 4.2 A Cataract Tele-Surgery

„The **D**istributed & **C**ollaborative **V**irtual **E**nvironments **R**esearch Laboratory (DISCOVER)“ der Universität Ottawa (Kanada) entwickelte auf der Grundlage von „**H**apto-**A**udio-**V**isual **E**nvironments for Collaborative Tele-Surgery Training over Photonic Networking (HAVE)“ eine Simulation der Kataraktoperation.

Das Projekt lief von 2005 bis 2006 und wird in [21] beschrieben. Der Aufbau gliedert sich in vier Teile, der grafischen Benutzeroberfläche (engl. *Graphical User Interface*, kurz GUI), der grafischen Visualisierung, der haptischen Kopplung mit Kollisionserkennung und -behandlung und der Netzwerkanbindung. Über die GUI lassen sich grundlegende Einstellungen vornehmen, wie z. B. die Auswahl der Instrumente und dem Blickwinkel. Die grafische Visualisierung ist für die Darstellung des Augmodells und der virtuellen Instrumente zuständig. Über haptische Eingabegeräte kann der Benutzer mit der Umgebung interagieren. Die physikalische Simulation basiert auf xPHEVE

[68], einer Eigenentwicklung, wobei zunächst nur ganz rudimentäre Möglichkeiten der Manipulation implementiert wurden und das Modell nur aus einer Oberflächensimulation besteht. Über die Netzwerkanbindung ist es möglich, passiv oder aktiv an der Simulation teilzunehmen.

## 4.3 EYESIM: Cataract Surgery Simulator

Am „Center for Advanced Studies, Research and Development in Sardinia (CRS4)“ (Italien) wurde ein Simulator für die Kataraktoperation entwickelt [2] und [1].

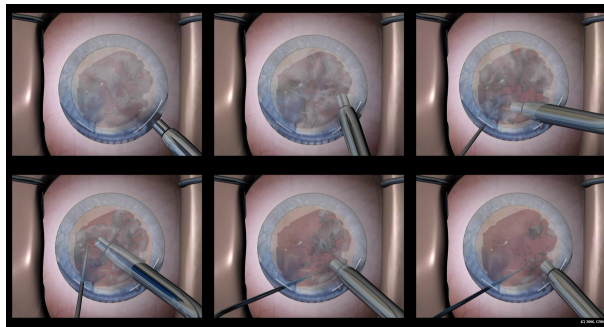


Abbildung 4.2: Screenshots vom EYESIM Simulator des CRS4. (Quelle: CRS4)

Das Projekt unter dem Namen „EYESIM: Cataract Surgery Simulator“ wurde vom 01.01.2005 über 15 Monate von der „DIES Group“ finanziert und gilt seitdem als abgeschlossen. In Abbildung 4.2 ist die Simulation der Phakoemulsifikation zu sehen.

Das System wurde ins DEVEYES System der DIES Group integriert und wird zum Training verwendet. Der erste Kurs wurde am 24. und 25.05.2006 im S. Raffaele Krankenhaus in Mailand durchgeführt.

Das System besteht aus zwei PCs, die über 100 Mbit Ethernet verbunden sind. Ein Rechner übernimmt die hochfrequente Trackingaufgabe (100 Hz), der andere die niederfrequente Simulationsaufgabe (20-25 Hz). Dabei werden das Auge, die Vorderkammer und die Linse simuliert. Des Weiteren besteht das System aus zwei Phantom Geräten zur haptischen Rückkopplung und einem N-Vision VB30 Stereo-Display. Das System ermöglicht eine Simulation der Einschnitte (*Incisionen*), der Kapsulorhexis und der Phakoemulsifikation. Bei der Simulation der Einschnitte ist es möglich, den Eintrittspunkt

zu wählen, der Schnitt selbst wird als Kreisbogen modelliert und die Ausmaße werden während dem Schnitt immer wieder neu berechnet.

Die Kapsel wird durch ein Feder-Masse-Modell modelliert. Instrumentinteraktionen werden durch Kollisionserkennung und deskriptive (geskriptete) Routinen erzielt. Für die numerische Integration wird ein implizites Verfahren verwendet. Am Ende jedes Zeitschritts werden die Partikelpositionen und -geschwindigkeiten an die physikalischen Beschränkungen angepasst. Das Reißen der Kapsel wird dadurch erzielt, dass die Federn getrennt werden, auf die die stärksten Kräfte wirken und die eine maximale Auslenkung überschritten haben. Die Rissausbreitung verläuft in Richtung der stärksten Spannung. Alle diese physikalischen Effekte basieren auf einer Vielzahl von Parametern, die angemessen eingestellt werden müssen, um ein realitätsnahes Verhalten zu erreichen.

Für die Phakoemulsifikation wurde eine Linse basierend auf einem Tetraedergitter (engl. *Mesh*) generiert. Zusätzlich wurde ein Massepunkt im Schwerpunkt jedes Tetraeders platziert. Die Bewegungen der Massepunkte werden mit einem Shape-Matching-Verfahren [63] simuliert. Der Phakoemulsifikationseffekt wird durch ein von ihnen „Russian roulette scheme“ genanntes Verfahren modelliert. In dies fließt die Entfernung, die Ultraschallstärke und die Kernhärte ein. Wie bei ihrer Simulation der Kapsulorhexis müssen auch bei dieser Simulation eine Reihe von Parametern sorgfältig gewählt werden, um ein realistisches Verhalten zu erzielen.

### 4.4 SOFA::EYE

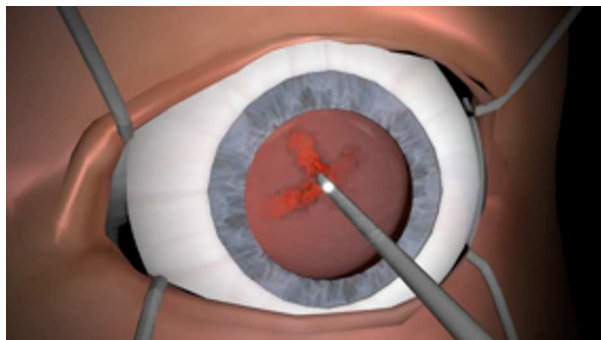


Abbildung 4.3: Screenshot vom SOFA::EYE<sup>©</sup> Simulator von INRIA (mit freundlicher Genehmigung von INRIA, <http://www.sofa-framework.org>).



Ende 2007 wurde am „Institut National de Recherche en Informatique et Automatique (INRIA)“ (Frankreich) mit der Entwicklung des Simulators SOFA::EYE, basierend auf der „Simulation Open Framework Architecture (SOFA)“, für die Kataraktoperation begonnen. Ihre Ziele sind die Simulation der Kapsulorhexis, der Phakoemulsifikation und das Einsetzen der Kunstlinse. Zurzeit beschränken sich die Informationen auf die Ankündigungen und einige Screenshots ähnlich der Abbildung 4.3. Aus dem verfügbaren Bild (Abb. 4.4) ist der genaue Aufbau nicht erkennbar.

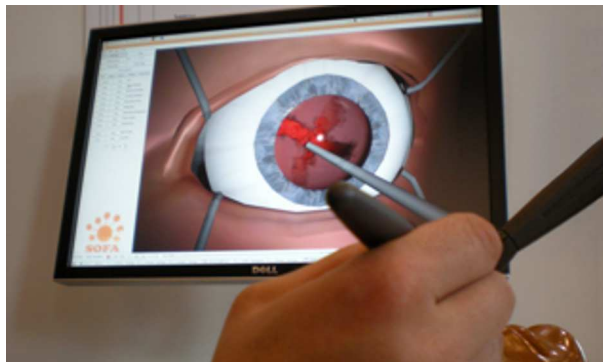


Abbildung 4.4: Der Aufbau des SOFA::EYE<sup>©</sup> Simulators von INRIA (mit freundlicher Genehmigung von INRIA, <http://www.sofa-framework.org>).

## 4.5 Eyesi

In diesem Abschnitt wird kurz auf den Augenoperationssimulator Eyesi (engl. *Eye Surgery Simulator*) eingegangen, da dieser den Rahmen im Bereich der Hard- und Software der vorliegenden Arbeit bildet.

Der Augenoperationssimulator Eyesi wurde von der Arbeitsgruppe ViPA (engl. *Virtual Patient Analysis*) an der Universität Mannheim entwickelt. In der Zwischenzeit wird er von der VRmagic GmbH weiterentwickelt und vertrieben.

Am Anfang der Entwicklung des Operationssimulators Eyesi stand das Training von Operationen am hinteren Augenabschnitt im Vordergrund. Während vorliegender Arbeit wurde Eyesi von der VRmagic GmbH, vor allem auf der Hardware-Seite, für Operationen am vorderen Augenabschnitt erweitert.

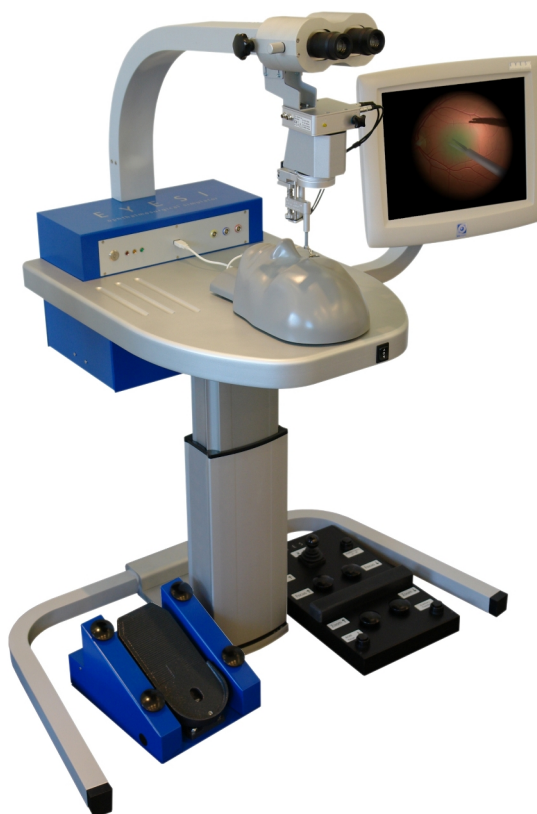


Abbildung 4.5: Der aktuelle Aufbau der Eyesi Plattform mit freundlicher Genehmigung der VRmagic GmbH.

### 4.5.1 Hardware

Der Aufbau des Operationssimulators orientiert sich streng an der realen Umgebung im Operationssaal und wird genauer in den Arbeiten [76], [99] und [32] beschrieben. Hier wird nur kurz auf den grundlegenden Aufbau eingegangen.

Um die reale Operationssituation so gut wie möglich nachzubilden, besitzt Eyesi (Abb. 4.5) ein Stereo-Display in Form eines Stereomikroskops, ein BIOM (**B**inokulares **I**ndirektes **O**phthalmomikroskop), einen Touchscreen, das haptische Operationsinterface, Instrument-Dummies und Fußpedale.

Im Stereo-Display wird die simulierte Operationsumgebung mit computergrafischen Mitteln dargestellt. Über den Touchscreen lässt sich das Gerät bedienen, des Weiteren wird er für die Darstellung der nachgebildeten Maschinen-Benutzeroberfläche verwendet. Eyesi lässt sich komplett über den Touchscreen bedienen, der auch für die Maschinen benutzt werden soll. Das haptische Operationsinterface ist ein dem Menschen nachgebildeter Kopf, unter dessen Schale sich ein mechanisches Auge und die gesamte Elektronik, die für das optische Tracking benötigt wird, befindet. Das mechanische Auge ist so gelagert, dass es wie beim Menschen bewegt werden kann. Die rückstellenden Kräfte der Muskel übernehmen Federn.

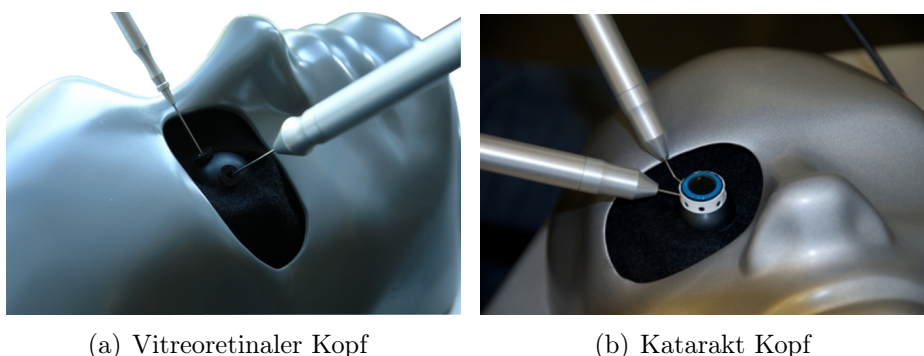


Abbildung 4.6: Die Eyesi Köpfe - Links die vitreoretinale Version und rechts die Katarakt Version mit freundlicher Genehmigung der VR-magic GmbH.

Bei chirurgischen Eingriffen am Auge unterscheidet man zwischen Operationen im „hinteren Bereich“ (vitreoretinal) und im „vorderen Bereich“ (Katarakt-Bereich). Zum „hinteren Bereich“ gehören alle Teile des Auges die sich hinter und zum „vorderen Bereich“ alle die sich vor der Linse befinden.

Da für Operationen an den verschiedenen Bereichen unterschiedliche Zugänge benötigt werden, gibt es zwei unterschiedliche Interfaces (siehe Abb. 4.6), eines für den vitreoretinalen und eines für den Katarakt-Bereich. Bei der vitreoretinalen Version erhält der Benutzer Zugang über zwei Bohrungen, die sich Pars plana zum Operationsgebiet befinden. Für Kataraktoperationen werden Zugänge auf der Höhe des Hornhaut-Lederhaut-Überganges benötigt. Dies wird in der mechanischen Nachbildung durch einen zylindrischen Aufsatz an dieser Stelle erreicht, der mit acht Bohrungen versehen ist, die zueinander jeweils einen Winkel von  $45^\circ$  besitzen.

In diese Bohrungen werden die Instrumenten-Dummies eingeführt und mit einem optischen Tracking verfolgt. Hierfür verfügt jedes Instrument über eine Farbmarkierung, die mit drei Kameras aufgezeichnet wird. Die Bilder der Kameras dienen als Grundlage für die Rekonstruktion der 3D-Position des Instruments, dessen Lage genau bestimmt ist durch die Farbmarkierung und das Einstichloch. Die vom Tracking ermittelten Objektpositionen werden dann in die virtuelle Welt übertragen. Über ein magnetisches Tracking der Instrumente wird ihre Rotation bestimmt. Das Magnetfeld wird durch Permanentmagnete, die um das mechanische Auge angebracht sind, erzeugt. Über einen Sensor im Handstück kann die Rotation des Instruments im Magnetfeld bestimmt werden.

Die ermittelten Objektkoordinaten dienen als Grundlage für die Position und Orientierung der computergrafischen Modelle, die ausschlaggebend für die Interaktion mit der virtuellen Realität sind.



Abbildung 4.7: Das Fußpedal von Eyesi zur Maschinensteuerung.

Zwei Fußpedale (siehe Abb. 4.5) dienen als weitere Eingabegeräte. Ein Fußpedal ist für die Bedienung des Mikroskops vorhanden. Der Benutzer kann

damit den Ausschnitt der virtuellen Szene verändern. Mit einem weiteren Fußpedal (siehe Abb. 4.7) ist die Steuerung der Operationsmaschine möglich. Eine zusätzliche Hardware ist das BIOM, es bildet eine spezielle Optik nach, mit der extreme Weitwinkelleinstellungen für den vitreoretinalen Bereich möglich sind.

Die aktuelle Eyesi-Spezifikation stammt aus dem Jahre 2004. Auch wenn sie in der Zwischenzeit an die aktuelle Technik angepasst wurde, muss jegliche für Eyesi entwickelte Software aus Kompatibilitätsgründen auf dieser Version lauffähig sein. Eyesi basiert auf Standard-PC-Komponenten und spezieller Hardware-Erweiterungen, die z. B. Sonderaufgaben wie das optische Tracking übernehmen.

Als Anhaltspunkt für die eingesetzte PC-Technik werden hier die Hauptkomponenten aufgeführt. Als Prozessor kommt ein Intel Pentium IV 2,8 GHz und als Hauptgrafikkarte eine ATI X800pro 256 MB zum Einsatz.

### 4.5.2 Software

Die Eyesi-Software baut auf der VRM-Architektur (engl. *Virtual Reality in Medicine*) auf. Eine weitreichende Beschreibung findet sich in [99]. Die VRM-Architektur beinhaltet Bibliotheken für die Simulation, die Kollisionserkennung und -behandlung, sowie weitere in der virtuellen Realität benötigten Teile. Sie ist in C++ programmiert und wird für Linux und Windows entwickelt. Als Entwicklungsumgebung dient Microsoft Visual Studio 2005. Für die Benutzeroberfläche kommt QT von Nokia zum Einsatz.

Der derzeitige Stand der Software-Bibliotheken soll hier nicht näher wiedergegeben werden, da diese ständig weiterentwickelt werden. Die Grundlage der aktuellen Version bilden die Konzepte aus den Arbeiten von [76], [99] und [32]. Neben den Grundlagen in Simulation und Visualisierung besitzt Eyesi einige spezielle Software-Eigenschaften. Diese sollen hier nicht unerwähnt bleiben. So wurde Eyesi als „geschlossenes“ System entwickelt. Das bedeutet, dass es komplett über den angebrachten Touchscreen gesteuert werden kann. Die Steuerung gliedert sich in zwei Bereiche: in den für Administratoren und den für Anwender. Über den Anwenderbereich ist die Auswahl der Trainingsaufgaben möglich. Die in Eyesi vorkommenden Trainingsaufgaben werden als *Trainingsmodule* bezeichnet. Es gibt zwei Arten von Trainingsmodulen, *abstrakte Trainingsmodule* und *Operationssimulationen*. Bei abstrakten Trainingsmodulen handelt es sich um einfache Aufgaben, mit denen der Lernende eine spezielle motorische Fertigkeit erlernen soll. Bei Operationssimulationen wird versucht einen Teil einer realen Operation oder eine komplette Operation so realistisch wie möglich zu modellieren. Dem Lernenden ist hierbei freigestellt wie er die Aufgabe löst.

Eine objektive Bewertung findet über das *Scoring* statt. Dabei werden Punkte für das Lösen der Aufgabe vergeben und für Fehler wieder abgezogen. Da für jedes Trainingsmodul Punkte vergeben werden, kann der Trainingsverlauf für jeden Benutzer protokolliert werden. Es ist ebenfalls möglich über den *Recorder* das ausgeführte Trainingsmodul noch einmal abzuspielen. Dieses kann dann z. B. von einem erfahrenen Chirurgen begutachtet werden. Eyesi wurde als Mehrbenutzersystem entwickelt. Die Benutzerverwaltung kann vom Administrator durchgeführt werden. Er nimmt die Einteilungen von Kursen und Mindestpunktzahlen zum Bestehen der Kurse vor. Da Eyesi zur chirurgischen Ausbildung eingesetzt wird, wurde viel Wert auf dieses Curriculum gelegt.

Während dieser Arbeit wurde die Eyesi-Software um einige Teile im Bereich Simulation, Kollisionserkennung und Trainingsmodule von dem Autor erweitert.

### 4.5.3 Anerkennung in der Ausbildung

Seit der Präsentation des Eyesi Prototypen im Jahre 2001 auf dem jährlichen Kongress der Deutschen Ophthalmologischen Gesellschaft (DOG), hat Eyesi die Art der Ausbildung wesentlich beeinflusst. Als Alternative zu den Wetlabs<sup>1</sup> entstanden die *Drylabs*. In einem Drylab wird ausschließlich am Simulator trainiert.

Auf dem 6. Internationalen Vitreoretinalen Symposium (2003) in Frankfurt-Marburg fand das erste Drylab statt. Ebenfalls im Jahre 2003 beschloss die DOG, das simulatorgestützte Training in die Ausbildung aufzunehmen. Für den jährlichen stattfindenden Kongress, werden seitdem Eyesi-Geräte gemietet. VRmagic veranstaltet seit 2006 das Europäische Vitreoretinale Drylab in Mannheim. Dieses wird von Augenärzten aus ganz Europa besucht. In den USA wurde die simulatorgestützte Ausbildung im Jahre 2007 in die Ausbildungsrichtlinien aufgenommen.

Die Bekanntheit des Systems und die Anerkennung als weiteres Standbein in der Ausbildung treiben die Entwicklung des Simulators immer weiter voran. Dazu gehören auch die Erweiterung für den vorderen Augenabschnitt sowie der Anbindung virtueller Operationsmaschinen, die hier erläutert werden sollen.

---

<sup>1</sup>Herkömmliche Ausbildung an präparierten Schweineaugen.

## **4.6 Abgrenzung der Arbeit**

Die meisten der erwähnten Projekte besitzen den Grad eines Prototypen oder einer Machbarkeitsstudie. Am weitesten fortgeschritten sind PhacoVision® und EYESIM. Diese wurden schon zur Ausbildung eingesetzt, aber auch sie befinden sich noch in einem frühen Entwicklungsstadium.

Keine der Arbeiten beschäftigt sich mit der Anbindung realer Operationsmaschinen und deren Simulation, wie es in dieser Arbeit der Fall ist. Des Weiteren werden Methoden für die Auswirkungen der virtuellen Operationsmaschine auf die gesamte virtuelle Augumgebung und die lokal eingesetzten Simulationsobjekte beschrieben.





## Die Operationsmaschine

Für die reale Durchführung einer Phakoemusifikation wird eine so genannte Phakomaschine oder Operationsmaschine benötigt. Um eine qualitativ hochwertige Ausbildung zu bieten, ist der korrekte Umgang mit dieser Operationsmaschine essentiell. Es ist ein Grundverständnis über die Physik der Phakoemulsifikation notwendig. Neben den theoretischen Grundlagen sollte der Operateur auch einen Überblick über die technische Umsetzung der Maschine erlangen. Dadurch ist es ihm möglich, die für die aktuelle Operation besten Einstellungen zu wählen.

Die Simulation der Phakoemulsifikation spielt sich in einer virtuellen Umgebung ab. Deshalb muss die Operationsmaschine und ihre physikalischen Eigenschaften in die virtuelle Umgebung übertragen werden.

Dafür wurde während dieser Arbeit ein Modell für die Abläufe in der virtuellen Operationsmaschine entwickelt. Obwohl sich derzeit weltweit einige Gruppen mit der Simulation der Phakoemusifikation beschäftigen, ist dem Autor keine Veröffentlichung bekannt, die sich speziell mit den Eigenschaften real existierender Maschinen und deren Übertragung in die virtuelle Umgebung beschäftigt.

Für die Modellbildung ist es wichtig, den Aufbau und die Abläufe innerhalb der realen Maschine zu kennen. Derzeit sind Maschinen von verschiedenen Herstellern auf dem Markt, jedoch besitzen sie alle gemeinsame Grundfunktionalitäten, die in der Fachliteratur [80] und [4] beschrieben werden.

Hier sollen zunächst die physikalischen Grundlagen der Phakoemulsifikation erläutert werden. Um dem Leser einen Überblick über die Phakomaschine zu geben, werden hier die Grundfunktionalitäten am Beispiel realer Maschinen beschrieben.

Zunächst wird der allgemeine Aufbau der Maschine beschrieben, dann auf die Integration in die virtuelle Realität eingegangen.

## 5.1 Physik der Phakoemulsifikation

Bei der Phakoemulsifikation wird die Linse mit Ultraschall verflüssigt und gleichzeitig aus dem Auge abgesaugt.

**Ultraschallerzeugung** Der Ultraschall wird im Phakohandgriff (Phakoinstrument) erzeugt. Es werden hier Standard-Phakohandgriffe wie in Abbildung 5.1 dargestellt betrachtet.

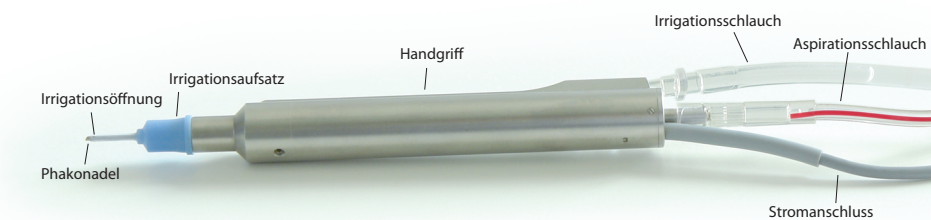


Abbildung 5.1: Das Phakohandstück, ein Modell von Geuder, mit Irrigations- und Aspirationsschlauch, Stromanschluss, Phakonadel und Irrigationsaufsatz.

Beim Phakoinstrument wird die Phakonadel (Phakolanz) an der Spitze des Instruments zum Schwingen angeregt. Dabei wird die Nadel in longitudinaler Richtung ausgelenkt. Dies kann durch magnetische oder piezoelektronische Anregung erfolgen. Die Nadel bewegt sich dabei im Bereich von wenigen  $\mu\text{m}$  mit einer nicht verstellbaren Frequenz im Ultraschallbereich von ca. 25 - 45kHz.

**Auswirkungen** Durch das Vibrieren der Nadel werden mehrere Effekte ausgelöst, die auf mikroskopischer Ebene für das Verflüssigen des Kerns verantwortlich sind.

Nach [12] lassen sich diese einteilen in:

- Nicht-Kavitation-Kräfte und mechanische Kräfte,
- Wärmeeffekte und
- Kavitation.

Zu den Nicht-Kavitation-Kräften und mechanischen Kräften zählen dabei die Stoßkräfte, der Strahlungsdruck, das *Acoustic streaming* und die Scherkräfte der Flüssigkeit. Die Wärmeeffekte beinhalten die Absorption der Schallenergie und das damit verbundene Aufheizen des Gewebes. Kavitation bezeichnet das Entstehen von Hohlräumen (Mikroblasen) in Flüssigkeiten durch Druckunterschied.

Die Mediziner sind sich uneins darüber, welcher Effekt hauptsächlich für das Verflüssigen verantwortlich ist. Das weiter unten vorgestellte Modell (siehe Abschnitt 5.4.4) berücksichtigt deshalb die Effekte des Ultraschalls auf makroskopischer und nicht auf mikroskopischer Ebene.

**Flüssigkeitsverhalten (engl. *fluidics*)** Während der Operation wird für mehrere Aufgaben Flüssigkeit benötigt: für den Abtransport des Materials aus der Vorderkammer, für die Kühlung des Instruments und für die Stabilität der Vorderkammer.

Das gesamte Flüssigkeitsverhalten wird als *Phaco Fluidics* oder *Phacodynamics* [80] bezeichnet. Es wird dabei zwischen Irrigation und Aspiration unterschieden. Irrigation bezeichnet den Zufluss ins Auge und Aspiration den Abfluss. Dabei wirkt sich die Irrigation hauptsächlich auf den Druck in der Vorderkammer aus. Die Aspiration ist für das Verhalten der Linsenstücke in der Vorderkammer verantwortlich.

**Vorbemerkungen** Im Folgenden werden zwei Begriffe verwendet: die *aktive Instrumenteigenschaft* und die *instrumentspezifische Eigenschaft*.

Die aktive Instrumenteigenschaft bezeichnet alle über die Maschine gesteuerten Auswirkungen des Instruments auf die reale und virtuelle Umgebung. Die instrumentspezifische Eigenschaft ist die Eigenschaft, die das Instrument auszeichnet. Beim Phakohandstück ist dies der Ultraschall, beim Vitrektor die Schnittrate.

## 5.2 Allgemeiner Aufbau der Operationsmaschine

Die einzelnen Eigenschaften werden nun am Beispiel einer einfachen fiktiven Maschine genauer erläutert, die sich an der aktuellen Generation der am Markt befindlichen Maschinen orientiert. Die Operationsmaschine erzeugt die für die Operation notwendigen aktiven Eigenschaften des Instruments. Gesteuert wird sie vom Chirurg.

Aktuelle Operationsmaschinen bestehen im Wesentlichen aus folgenden Teilen:

- dem Touchscreen,
- dem Fußpedal,
- dem Infusionsständer (Flaschenhalter),
- den Pumpen,
- den Anschlüssen und den Schläuchen (Schlauchsystem),
- der für die instrumentspezifische Eigenschaften zuständigen Steuerung,
- der internen Maschinenregelung.

Die spezielle Umsetzung ist den Maschinenherstellern vorbehalten und kann sehr unterschiedlich sein. Einen knappen Überblick über drei aktuell am Markt verfügbare Maschinen gibt [71]. Für die Modellbildung wurden nur Maschinen, die über eine Touchscreen-Steuerung verfügen, betrachtet. Diese ermöglichen eine einfachere Übertragung der eingestellten Werte in die virtuelle Realität.

Abbildung 5.2 zeigt in alphabetischer Reihenfolge eine Auswahl aktueller Phakomaschinen von Alcon, AMO, DORC und Geuder. (Es ist eine alphabetische Reihenfolgen gewählt worden, um nicht den Eindruck zu erwecken, dass eine Maschine bevorzugt würde. Gleiches gilt für die Anordnung der nachfolgenden Bilder.)

Reale Maschinen wie in Abbildung 5.2 können nicht direkt in der virtuellen Realität benutzt werden. Sie müssen in das virtuelle System integriert und die eingestellten Werte müssen übertragen werden. Um die Werte, die in die virtuelle Realität übertragen wurden, verarbeiten zu können, wurde die Operationsmaschinen-Simulation (OMS) entwickelt. Bei der Entwicklung der OMS wurde darauf geachtet, die Eigenschaften einer realen Operationsmaschine so allgemein wie möglich abzubilden.

Um eine allgemeine virtuelle Operationsmaschine modellieren zu können, müssen der Aufbau und die Abläufe einer realen Maschine analysiert werden.

Im Folgenden werden die oben aufgeführten Komponenten näher betrachtet.

### 5.3 Die Komponenten

**Touchscreen** Ein zentraler Punkt der aktuellen Maschinengeneration ist der Touchscreen. Über ihn werden alle Werte der Maschine, wie z.B. die



(a) Das Infiniti® Vision System mit freundlicher Genehmigung der Alcon Laboratories, Inc.



(b) Die AMO Gemini mit freundlicher Genehmigung von Abbott Medical Optics Inc.



(c) Die AMO WHITEST-AR Signature™ mit freundlicher Genehmigung von Abbott Medical Optics Inc.



(d) Die D.O.R.C. Associate® mit freundlicher Genehmigung der D.O.R.C. Deutschland GmbH

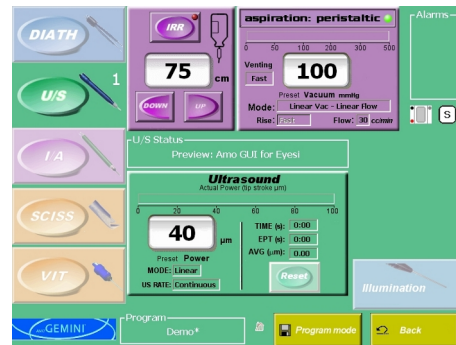


(e) Die Geuder Megatron-S3 mit freundlicher Genehmigung der GEUDER AG

Abbildung 5.2: Eine Auswahl aktueller Phakomaschinen.



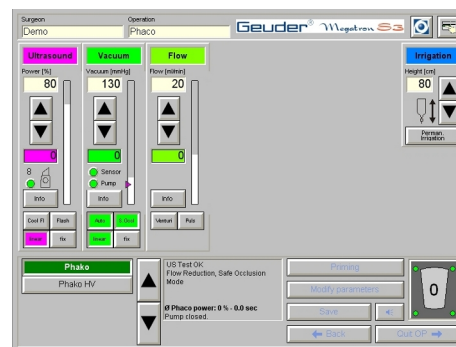
(a) Die Infiniti® Vision System Benutzeroberfläche mit freundlicher Genehmigung der Alcon Laboratories, Inc.



(b) Die AMO Gemini Benutzeroberfläche mit freundlicher Genehmigung von Abbott Medical Optics Inc.



(c) Die AMO WHITESTAR Signature™ Benutzeroberfläche mit freundlicher Genehmigung von Abbott Medical Optics Inc.



(d) Die Geuder Megatron S3 Benutzeroberfläche mit freundlicher Genehmigung der GEUDER AG.

Abbildung 5.3: Einige Benutzeroberflächen aktueller Phakomaschinen.

Aspiration und die Irrigation konfiguriert. Zusätzlich erhält der Chirurg Informationen zu den aktuellen Werten.

Die in Abbildung 5.3 dargestellten Benutzeroberflächen sind eine Auswahl der aktuell erhältlichen Maschinen. Es ist jeweils der Zustand während der Operation dargestellt. Über ihn können die Eigenschaften der Maschine manipuliert werden.

Neben den Einstellungen für Irrigation, Aspiration und die Ultraschallleistung bieten die Maschinen weitere unterschiedliche Einstellungsmöglichkeiten. Die einzelnen Werte werden weiter unten genauer erläutert.

Über die graphische Benutzeroberfläche auf dem Touchscreen stehen weitere Möglichkeiten, wie Verwaltungsaufgaben, zur Verfügung. Auf diese wird hier nicht weiter eingegangen.

**Fußpedal** Während der Operation benötigt der Chirurg seine Hände für die Operationsinstrumente. Mit den Füßen bedient er die Maschine mit einem Fußpedal, ähnlich den in Abb. 5.4 dargestellten. Fußpedale stehen in verschiedenen Ausführungen, mit mehr oder weniger Freiheitsgraden, zur Verfügung. Alle Fußpedale besitzen zwei Arten von Eingabegeräten, digitale Schalter und analoge Wegaufnehmer (vergleichbar mit einem Gaspedal). Für die analogen Wegaufnehmer gibt es zwei gebräuchliche Varianten, die *Einfach-Linearen* und die *Zweifach-Linearen*. Die Einfach-Linearen besitzen nur einen Freiheitsgrad in vertikaler Richtung. Die Zweifach-Linearen sind um einen Freiheitsgrad in horizontaler Richtung erweitert.

Mit den analogen Eingabegeräten am Fußpedal werden die Irrigation, die Aspiration und die instrumentspezifische Eigenschaft gesteuert. Jede Eigenschaft ist einem Bereich auf dem Fußpedal zugeordnet. Im Folgenden werden die Bereiche entsprechend der vorherigen Reihenfolge als **A**, **B** und **C** bezeichnet.

Für Einfach-Lineare Fußpedale (siehe Abb. 5.5) gilt, dass die Bereiche **A** bis **C** hintereinander auf einer Achse angeordnet sind. Die Steuerung der Effekte erfolgt additiv. Bei Zweifach-Linearen (siehe Abb. 5.6) Pedalen erfolgt die Steuerung der instrumentspezifischen Eigenschaft auf einer weiteren Achse und ist somit unabhängig von der Irrigation und der Aspiration.

Eine weitere Einstellungsmöglichkeit an der Maschine ist der **Linear-** oder **Fix-Betrieb**. Beim Linear-Betrieb verhält sich das Fußpedal wie ein Gaspedal, je stärker es gedrückt ist, desto höher ist der Wert und er nähert sich linear dem eingestellten Wert. Im Fix-Betrieb wird dieser vorgegebene Wert sofort erreicht. Für die Aspiration kann eine Anstiegszeit (engl. *rise time*) vorgegeben werden. Der aktuelle Wert nähert sich dann dem eingestellten Wert über eine definierte Zeitspanne an.



(a) Das Infiniti®Vision System Fußpedal mit freundlicher Genehmigung der Alcon Laboratories, Inc.



(b) Das AMO WHITES-TAR Signature™ Fußpedal mit freundlicher Genehmigung von Abbott Medical Optics Inc.



(c) Das D.O.R.C. Fußpedal mit freundlicher Genehmigung der D.O.R.C. Deutschland GmbH.



(d) Das Geuder Fußpedal mit freundlicher Genehmigung der GEUDER AG.

Abbildung 5.4: Einige Fußpedale aktueller Phakomaschinen.



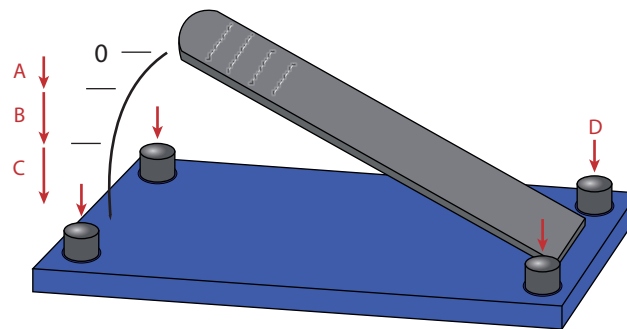


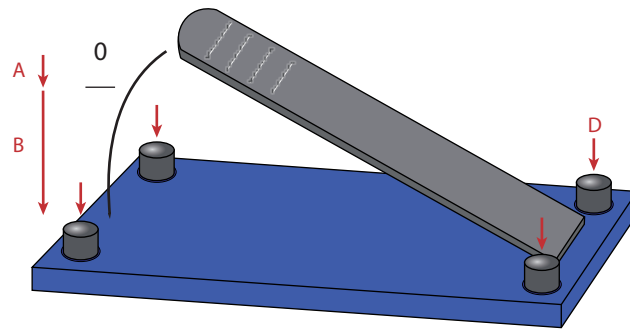
Abbildung 5.5: Die Schaltpunkte **A** bis **C** eines Einfach-Linearen Fußpedals mit 4 digitalen Schaltern, wie z. B. **D**.

**Infusionsständer** Am Infusionsständer wird die Infusionsflasche für die Irrigation angebracht. Die Irrigation selbst wird passiv durch die Infusionsständerhöhe gesteuert. Über eine *Tropfkammer* wird dabei ein füllstandsunabhängiger Schweredruck erzeugt, der den Zufluss bestimmt. Die Infusionsflasche enthält die für die Operation benötigte BSS-Lösung (engl. *Balanced Salt Solution*). Über ein schnell reagierendes Ventil an der Maschine wird die Irrigationsleitung geöffnet oder geschlossen.

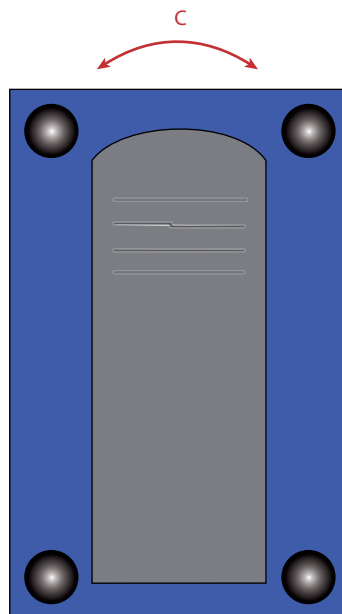
**Pumpen** Die Aspiration wird aktiv über eine Pumpe erzeugt. Dabei kommen zwei verschiedene Pumpentypen zum Einsatz: *Venturi*- und *Peristaltik*-Pumpen. Nach ihrem Arbeitsprinzip nennt man Venturi-Pumpen auch Vakuumumpen (Strahlpumpen) und Peristaltik-Pumpen auch Flusspumpen.

Es gibt viele verschiedene Möglichkeiten die Pumpen zu realisieren. Exemplarisch wird für jeden Pumpentyp eine Realisierungsmöglichkeit betrachtet.

**Flusspumpen** Flusspumpen arbeiten nach dem Peristaltik-Prinzip. Bei den in Phakomaschinen eingesetzten Typen handelt es sich meist um so genannte Rollenpumpen. Der Aspirationsschlauch wird zwischen einer festen Wand und einer Rolle eingeklemmt. Diese Rolle hat Erhebungen, die den Schlauch abklemmen und die Flüssigkeit vor sich herschieben, wenn der Pumpenkopf rotiert. Ähnlich wie beim Melken kommt so eine Fließbewegung zustande. In Abbildung 5.7 wird der exemplarische Aufbau einer Rollenpumpe dargestellt. Damit die Pumpe gut arbeiten kann, darf die Wandstärke des Schlauches nicht zu groß sein, da der Widerstand sonst zu hoch wäre. Dabei wird ein direkter Fluss in der Aspirationsleitung erzeugt. Der dabei zu überwindende Leitungswiderstand wird als Vakuum (negativer Druck in Bezug auf die



(a) Seitenansicht



(b) Draufsicht

Abbildung 5.6: Die Schaltpunkte **A** und **B** auf der horizontalen und der Schaltbereich **C** auf der vertikalen Achse eines Zweifach-Linearen Fußpedals mit 4 digitalen Schaltern, wie z. B. **D**.

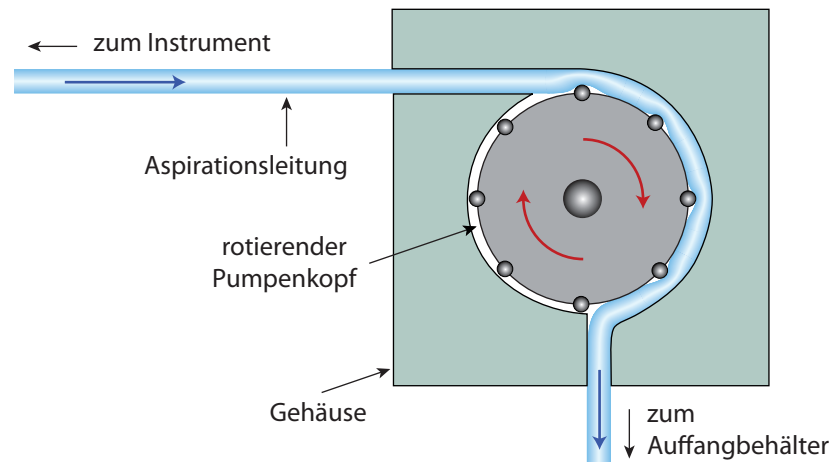


Abbildung 5.7: Die Peristaltik-Pumpe als schematische Darstellung. In der Mitte befindet sich der rotierende Pumpenkopf, der die Flüssigkeit durch den Schlauch presst und dabei den Fluss erzeugt.

Umgebung) im Vakuumsensor, der sich in der Nähe der Pumpe befindet, gemessen. Bei einer Verstopfung der Phakospitze steigt das Vakuum an, bis es den voreingestellten Grenzwert erreicht, die Pumpenkopffrotation kommt dabei zum Erliegen. Der Fluss sowie ein begrenzendes Vakuum sind über den Touchscreen einstellbar. Das Vakuum in der Leitung entwickelt sich über eine gewisse Anstiegszeit. Die Anstiegszeit ist umgekehrt proportional zur Pumpenkopffrotation.

**Vakuumpumpen** Im Gegensatz dazu wird bei den Vakuumpumpen ein direktes Vakuum erzeugt. Dies kann durch Ausnutzung des Venturi-Effektes<sup>1</sup> erfolgen. In Abbildung 5.8 ist der exemplarische Aufbau einer Venturi-Pumpe dargestellt. Die Aspirationsleitung wird an eine feste, nicht deformierbare Tropfkammer angeschlossen. In dieser wird ein Vakuum erzeugt, das die Flüssigkeit aus der Leitung zieht. Dabei wird zur Vakuumerzeugung der Venturi-Effekt ausgenutzt. Der Venturi-Effekt beschreibt das Verhalten von strömenden Gasen durch ein Rohr mit sich verengendem Querschnitt. An der Stelle mit dem dünnsten Querschnitt ist der dynamische Druck am stärksten und der statische Druck am geringsten. Wird an der dünnsten Stelle ein Abnehmerrohr angebracht, erzeugt der dynamische Druck einen Sog im Abnehmerrohr. Dieser Sog wird benutzt um die Tropfkammer zu evakuieren. Über die Anstiegszeit kann die Dauer, bis das zu erreichende Vakuum

<sup>1</sup>benannt nach Giovanni Battista Venturi

aufgebaut wurde, eingestellt werden. Das aktuell herrschende Vakuum wird in der Tropfkammer gemessen. An der Maschine wird es aus Gründen der Lesbarkeit in mmHg ohne Vorzeichen angegeben, obwohl es sich um einen Unterdruck handelt.

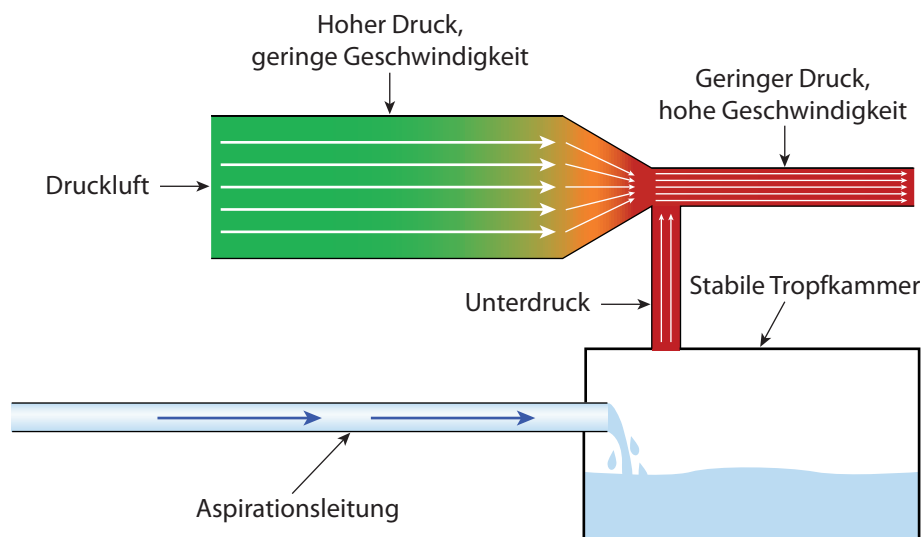


Abbildung 5.8: Die Venturi-Pumpe als schematische Darstellung. Die Druckluft strömt durch ein sich verengendes Rohr. Dabei entsteht am Abnahmerohr ein Unterdruck, der genutzt wird um die stabile Tropfkammer zu evakuieren. Der Unterdruck in der Tropfkammer bewirkt den Fluss der Flüssigkeit.

Der in der Aspirationsleitung herrschende Druckunterschied ist bei diesem Prinzip für den Fluss verantwortlich. Kommt es hierbei zu einer Verstopfung der Nadel wird der Druck direkt weitergegeben. An der Maschine kann nur der Vakuumgrenzwert eingestellt werden. Der Fluss ergibt sich indirekt und kann nicht kontrolliert werden. Phakomaschinen mit Vakuumpumpen werden allgemein hin als etwas aggressiver in Bezug auf ihr Absaugverhalten angesehen. Dies ermöglicht eine schnellere, aber auch riskantere Arbeitsweise.

**Anschlüsse und Schlauchsystem** Jede Maschine verfügt über eine Reihe von Anschlüssen für die Instrumente. Daran werden die Schläuche für die Flüssigkeit und die Leitungen für die aktive Instrumenteigenschaft angeschlossen. Das Schlauchsystem wird unterteilt in Irrigations- und Aspira-

tionsleitungen. Es hat eine nicht unerhebliche Auswirkung auf das Flüssigkeitsverhalten der Maschine. Die Zuleitung zum Auge wird als Irrigationsleitung bezeichnet. Diese besitzt einen großen Querschnitt und eine dünnere Wand. Dadurch wird sichergestellt, dass genug Flüssigkeit ins Auge nachfließen kann. Als Aspirationsleitung wird die Abflussleitung aus dem Auge bezeichnet. Sie hat einen kleineren Querschnitt und eine dickere Wand und ist dadurch stabiler und weniger anfällig für Deformationen. Dies kann beim Peristalik-Pumpentyp nur bedingt umgesetzt werden, da sonst die Funktionsweise nicht mehr gewährleistet ist. Die Schlauchlänge wirkt sich über den Strömungswiderstand auf das gesamte Strömungsverhalten aus. Da es sich um flexible Schläuche handelt, sind diese mit einer mehr oder minder starken *Compliance* behaftet. Die Compliance gibt den Zusammenhang zwischen Druck und Volumen bei umwandeten Strukturen an. Kommt es zu einer Verstopfung an der Instrumentspitze, baut sich ein Vakuum auf. Dies führt zu einer Verringerung des Volumens des Schlauchsystems und wird als Verformungsenergie gespeichert. Löst sich nun die Verstopfung, wird diese Energie freigesetzt und der Schlauch nimmt seine ursprüngliche Form an. Das Volumen vergrößert sich schlagartig und zieht dabei sehr viel Flüssigkeit aus dem Auge ab, es kommt zum Surge [80] und [18]. Der Surge ist die Ursache von Kapselrissen, Kammerkollaps oder Iriskollaps und muss auf jeden Fall vermieden werden.

Neben den geeigneten Maschineneinstellungen und einem sorgsamem Umgang bei der Steuerung, gibt es verschiedene zusätzliche Möglichkeiten zur Surge-Unterdrückung, diese sind z. B.:

- Aspirationleitungen mit dünnerem Durchmesser und größerer Wandstärke [18],
- ABS-Tips (engl. *Aspiration Bypass System*) mit zusätzlichen Bohrungen im Schaft (von Alcon),
- Externe Filter, wie der Cruise Control der Sonic Wave<sup>TM</sup> von STAAR Surgical,
- Spezielle Regelungstechniken, z. B. CASE (engl. *ChAmber Stabilization Environment*) bei AMO SOVEREIGN<sup>®</sup> COMPACT und AMO WHI-TESTAR Signature<sup>TM</sup>.

**Die Instrumente** Als letztes Glied in der Irrigationsleitung bzw. als erstes in der Aspirationsleitung hat das Instrument Auswirkungen auf das Strömungsverhalten. Das Instrument besitzt in der Kette der flüssigkeitsführenden Elemente den kleinsten Querschnitt und ist somit der restriktivste Teil für den Volumenstrom. Die Öffnungen können sich je nach Grad der Verstopfung verringern.

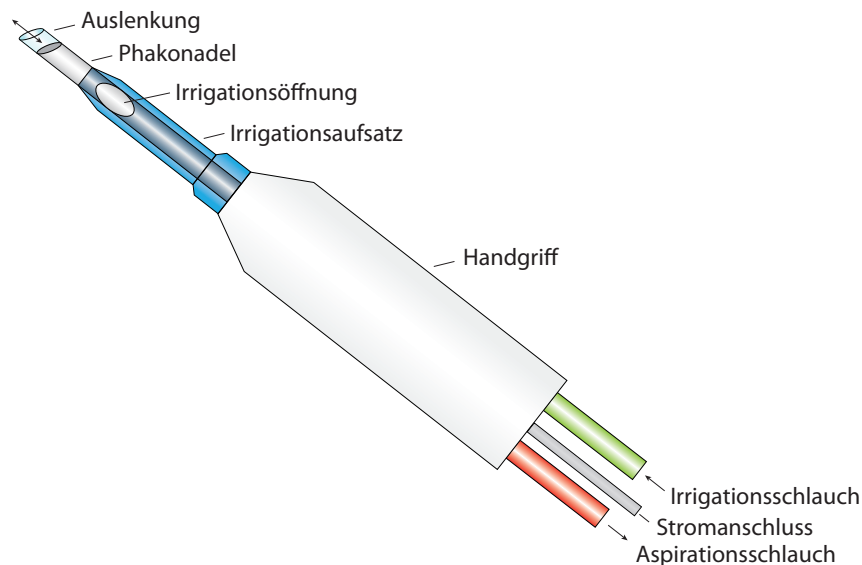


Abbildung 5.9: Das Phakohandstück - Schematisch, mit Irrigations- und Aspirationsschlauch, Stromanschluss, Phakonadel und Irrigationsaufsatz.

**Phakohandstück** Der Ultraschall, der durch das Phakohandstück (siehe Abb. 5.9) erzeugt wird, ist bestimmt durch die longitudinale Auslenkung der Nadel sowie die Frequenz. Die Auslenkung bewegt sich in einem Bereich von 0 bis wenigen  $100\ \mu\text{m}$ . Der eingesetzte Frequenzbereich liegt zwischen 27 und 54 kHz. Der Frequenzbereich ist von Maschine zu Maschine verschieden und an der Maschine nicht verstellbar. Es ist lediglich die Auslenkung manipulierbar. Bei dem Standardhandstück handelt es sich um ein koaxiales Instrument. Dabei erfolgt Irrigation und Aspiration über dasselbe Instrument. Dies wird durch einen Irrigationsaufsatz (engl. *irrigation sleeve*) aus Silikon erreicht, der über die Phakonadel gestülpt wird. Er besitzt zwei Öffnungen, durch die die Irrigationströmung strömen kann. Dieser Durchfluss dient zugleich als Kühlung für das Handstück.

Neben der Auslenkung kann auch die Modulation des Ultraschalls gesteuert werden. Der Ultraschall wird nicht dauerhaft aktiviert, sondern nach einem vorgegebenen Schema. Dafür stehen Standardmodi wie Pulse Mode, Hyperpulse und Burst Mode zur Verfügung. Dies reduziert den erzeugten Ultraschall. Die meisten Maschinenhersteller bieten weitere eigene Modi an.

Unter *Torsional Phaco* ist eine weitere Neuerung von Alcon bekannt geworden, bei der das Instrument nicht nur longitudinal ausgelenkt wird, sondern auch um die Instrumentachse schwingt. Dies soll die Repulsion an der In-

strumentspitze deutlich reduzieren.

**Interne Maschinenregelung** Die interne Maschinenregelung verarbeitet neben den Benutzereingaben, die über den Touchscreen und das Fußpedal getätigt werden, auch Werte, die über Druck- und Flusssensoren bereitgestellt werden. Über einen Regelkreis werden diese Werte verarbeitet und die aktuellen Werte angepasst.

Die Hauptaufgabe ist dabei die Druck- und Flusssteuerung, damit während der Operation stets eine stabile Vorderkammer erhalten bleibt und das Komplikationsrisiko minimiert wird.

Als Beispiel für solch ein Regelsystem kann das *Venting* genannt werden. Dabei wird Flüssigkeit oder Luft in die Aspirationsleitung eingebracht, um den Unterdruck in der Leitung schnell zu minimieren.

Die Entwicklung bei den Maschinenherstellern geht dabei zu speziellen Lösungen [71] für bestimmte Situationen. So reduzieren manche Maschinen automatisch das Vakuum, wenn es über einen zu langen Zeitraum auftritt.

## 5.4 Die Maschinen-Simulation

Um die Maschine in der virtuellen Realität verwenden zu können, müssen entsprechende Modelle entwickelt werden, die die reale Maschine in die virtuelle Welt übertragen. Auf die Integration der Maschine in die bestehende Eyesi Systemarchitektur wird in *Kapitel 7 - Software-Architektur* eingegangen.

**Berechnung der Maschinen-Werte** Für die Operationssimulation werden die Werte an der Instrumentspitze benötigt, die in der Realität von der Maschine erzeugt werden. Dafür wurde eine geeignete generalisierte Maschinen-Simulation entwickelt.

**OMS** Die OMS (Operationsmaschinen-Simulation) bildet das reale zeitliche Verhalten einer Operationsmaschine in der virtuellen Umgebung nach. Sie berücksichtigt dabei die physikalischen Eigenschaften der verwendeten Komponenten und die internen Regelmechanismen der realen Maschine. Sie stellt alle Werte, die für die Druck- und Volumenberechnung in der Vorderkammer benötigt werden und die instrumentspezifischen Eigenschaften zur Verfügung.

Maschinen-Wert	Üblicher Wertebereich	Einheit
Lichtquelle	0-100	Prozent
Aspirationsflussrate	0-50	ml/Minute
Aspirationsvakuum	0-1000	mmHg
Irrigationsdruck	0-100	cmH <sub>2</sub> O
Ultraschallstärke	0-100	$\mu$ m
Ultraschallpulserate	0-20	Pulse/Sekunde
Vitrector Cutrate	0-2000	Schnitte/Minute

Tabelle 5.1: Typische Wertebereiche für die einzelnen Parameter aktueller Maschinen.

Dies beinhaltet die Modellbildung für:

- die Steuerung der Irrigation über die Flaschenhöhe und die Öffnung des Irrigationsventils,
- die Steuerung der Aspiration über die Pumpe,
- die Berücksichtigung der Compliance der Schläuche,
- die Ultraschallstärke des Phakohandstücks und
- die Schnittrate des Vitrektors.

**Soll- und Ist-Werte** Bei der OMS wird zwischen Soll- und Ist-Werten unterschieden. Als Soll-Werte werden die direkt an der Maschine eingestellten Werte bezeichnet. Ist-Werte sind die von der OMS berechneten Werte, die in der Simulation benötigt werden.

Tabelle 5.1 gibt einen Überblick über die in der Simulation benötigten Werte. Es sind die in der Medizin üblichen Einheiten aufgeführt, für die Berechnungen in der OMS werden diese auf SI-Einheiten umgerechnet.

Als Beispiel für die Berechnung eines Ist-Wertes soll die Lichtquelle betrachtet werden. Bei einer Lichtquelle kommt es zu keiner Rückkopplung. Der an der Maschine eingestellte Wert kann direkt umgesetzt werden. In der virtuellen Realität kann dieser Wert uneingeschränkt benutzt werden, um z. B. als Intensitätswert für eine OpenGL-Lichtquelle dienen.

Der anspruchsvollste Teil ist die Modellierung des Flüssigkeitsverhaltens. In dessen Modellierung sind neben den physikalischen Grundlagen der Hydrodynamik auch Informationen verschiedener Maschinenhersteller und Chirurgen eingeflossen.

Es wurde versucht, das Modell auf einer fundierten physikalischen Grundlage aufzubauen. Jedoch ist das System sehr komplex, so dass an einigen Stellen nur die Möglichkeit der deskriptiven Modellierung zur Lösung führte.



**Vorbemerkungen** In der Realität handelt es sich um ein gekoppeltes System aus Irrigation, Vorderkammer und Aspiration. In den hier entwickelten Modellen bilden Irrigation und Aspiration unabhängige Teile. Die berechneten Werte sind die Basis für die Simulation der Vorderkammer, die im *Kapitel 6 - Gewebeinteraktion mit dem Instrumentarium* beschrieben wird.

Grundlage für die Flüssigkeitsberechnungen sind die Gleichungen der Strömungslehre mit deren Hilfe die Abhängigkeit von Druck und Fluss von Flüssigkeitsströmen durch Leitungen berechnet werden kann.

Des Weiteren werden als Vereinfachungen angenommen:

- laminaren Strömungen,
- kreisrunde Leitungsquerschnitte,
- konstante Reibungsverluste in den Leitungen,
- konstanter Druckverlust über die Leitungslänge.

Diese Annahmen beruhen auf Mess- und Erfahrungswerten, auf Rücksprache mit Maschinenherstellern, Ärzten und der Recherche in der Fachliteratur.

Im Folgenden werden die Werte, die über die Maschinen-GUI eingestellt werden mit dem Zusatz „GUI“, die Werte, die in der Maschine gelten mit dem Zusatz „Maschine“ und die die am Instrument gelten mit „Instrument“ bezeichnet. Die Maschinen-Werte entsprechen den Soll-Werten und die Instrument-Werte den Ist-Werten.

### 5.4.1 Fußpedal

Für die meisten Berechnungen wird die Stellung und die Konfiguration des Fußpedals benötigt. Dies wird in dem normierten Fußpedalwert FPW zusammengefasst. Für den Linear-Betrieb nimmt FPW Werte zwischen 0 und 1 an und für den Fix-Betrieb entweder den Wert 0 oder 1. Die einzelnen Bereiche (**A**, **B**, **C**) können unabhängig voneinander konfiguriert werden. Im weiteren Verlauf der Arbeit wird der FPW entsprechend seiner Konfiguration verwendet. Die genaue Umsetzung wird im *Kapitel 7 - Software-Architektur* erläutert.

### 5.4.2 Irrigation

Die Irrigation wird passiv über die relative Flaschenhöhe in Bezug auf die Patientenkopfhöhe erzeugt und über ein schnell reagierendes Ventil an- bzw. abgeschaltet. Eine Änderung der Flaschenhöhe erfolgt durch herauf- bzw.

herunterfahren des Infusionsständers über einen Motor mit konstanter Geschwindigkeit. Für die Höhenänderung gilt:

$$\Delta h(t)_{\text{Maschine}} = \begin{cases} m * t, & \text{wenn } h(t)_{\text{Maschine}} < h_{\text{GUI}} \\ 0, & \text{wenn } h(t)_{\text{Maschine}} = h_{\text{GUI}} \\ -m * t, & \text{wenn } h(t)_{\text{Maschine}} > h_{\text{GUI}} \end{cases} \quad (5.1)$$

mit der Motorgeschwindigkeit  $m$  und der relativen Zeit  $t$ , die beim Erreichen des Soll-Wertes zurückgesetzt wird.

Für die aktuelle Höhe gilt:

$$h(t)_{\text{Maschine}} = h_{\text{Maschine}}^0 + \Delta h(t)_{\text{Maschine}} \quad (5.2)$$

wobei  $h_{\text{Maschine}}^0$  der letzte gültige Maschinenwert vor der Änderung auf der GUI war. Entspricht die aktuelle Höhe dem GUI-Wert wird  $h_{\text{Maschine}}^0$  auf diesen neuen Wert gesetzt.

Für die Berechnung des Irrigationsdrucks  $p_{\text{Instrument}}$  wird der durch die Flaschenhöhe erzeugte Schweredruck mit dem Öffnungswert für das Ventil multipliziert. So erhält man durch Änderung des Ventils den Irrigationswert an der Instrumentenspitze:

$$p_{\text{Instrument}} = g\rho h(t)_{\text{Maschine}} * \delta_{\text{Ventil}} \quad (5.3)$$

Dabei ist  $h(t)$  die aktuelle Höhe in Bezug auf die Höhe des Patientenauges, der sich bei einer Änderung des Soll-Wertes über die Motorgeschwindigkeit errechnet.  $\delta_{\text{Ventil}}$  kann nur 0 oder 1 annehmen. Dies entspricht einem Fix-Betrieb des Fußpedals für die Irrigation. Der so errechnete Wert wird in der Vorderkammersimulation verwendet.

**Freier und gekoppelter Fluss** Bei freiem Fluss in die Umgebung (Luft) lässt sich der Fluss über den Schweredruck, der sich über die Flaschenhöhe ergibt, berechnen. Nach dem Gesetz von Hagen-Poiseuille (Gleichung 5.4) gilt somit für den ungehinderten Fluss  $\dot{V}$  durch die Instrumentenspitze:

$$\dot{V} = \frac{\pi r^4}{8\eta l} \Delta p \quad (5.4)$$

mit dem Radius  $r$  des Rohres, der Viskosität  $\eta$ , dem Druckunterschied  $\Delta p$  zwischen Anfang und Ende des Rohres und der Länge  $l$ .

Dies ist der theoretisch maximale Fluss an der Instrumentspitze. Durch Einbringen des Instruments in die Vorderkammer entsteht ein gekoppeltes System. Der Druck  $\Delta p$  an der Instrumentspitze reduziert sich um den aktuellen Druck in der elastischen Vorderkammer. Die Auswirkungen der elastischen Vorderkammer werden in *Kapitel 6 - Gewebeinteraktion mit dem Instrumentarium* untersucht. Die Maschine stellt der virtuellen Realität nur den aktuellen Schweredruck zur Verfügung. Die Berechnungen für den Irrigationsfluss werden in der virtuellen Realität vorgenommen.

### 5.4.3 Aspiration

Im Gegensatz zur Irrigation wird die Aspiration aktiv durch eine Pumpe erzeugt. Dabei wird zwischen zwei Werten unterschieden, dem Fluss und dem Vakuum. Die verwendete Pumpe bestimmt das Fluss- und Vakuumverhalten an der Instrumentspitze, dies wird bei einer Verstopfung besonders deutlich. Bei einer Peristaltik-Pumpe wird der Widerstand, gegen den sie arbeiten muss, immer stärker. Dies kann zum Stillstand der Pumpe führen. Bricht die Verstopfung auf, setzt der Fluss schlagartig ein und kann durch die Auswirkungen der Compliance noch verstärkt werden. Bei einer Venturi-Pumpe entsteht bei einer Verstopfung sehr schnell ein großer Unterdruck in der Leitung, der durch das Vakuum in der stabilen Tropfkammer erzeugt wird. Über die rise time ist die Zeit, die die Maschine benötigt, um das voreingestellte Sollvakuumwert in der Tropfkammer aufzubauen, einstellbar. Als Maßnahme gegen erhöhten Fluss oder erhöhtes Vakuum kommt Venting zum Einsatz. Beim Venting wird Flüssigkeit oder Luft in die Leitung eingeleitet, um den erhöhten Fluss zu verringern.

Für die Simulation in der virtuellen Realität werden die Aspirationswerte an der Instrumentspitze benötigt. Diese werden von mehreren Komponenten der Maschine beeinflusst: dem verwendeten Pumpentyp, dem Schlauchsystem und der damit einhergehenden Compliance, der Instrumentgeometrie und der Maschinenregelung mit rise time und Venting.

Im Folgenden werden die entwickelten Modelle vorgestellt.

#### 5.4.3.1 Peristaltik-Pumpe

Beim Modell der Peristaltik-Pumpe wird davon ausgegangen, dass der in der Pumpe erzeugte Fluss dem Fluss am Maschinenanschluss entspricht. Der Vakuumsensor in der Maschine misst dabei das Vakuum, das benötigt wird, um den Leitungswiderstand zu überwinden. Eine Verstopfung reduziert den Fluss an der Instrumentspitze und somit erhöht sich das gemessene Vakuum. Bei vollständiger Verstopfung kommt der Fluss an der Instrumentspitze zum

Erliegen. Die Pumpe arbeitet solange weiter, bis das eingestellte Vakuum erreicht wird. Die Flussdifferenz zwischen den beiden Enden der Aspirationsleitung führt zu einer Deformation des Schlauches. In dieser Deformation wird Energie gespeichert.

Für die Peristaltik-Pumpe werden in der OMS folgende Werte berechnet, die die Grundlagen für weitere Berechnungen bilden.

**Der Fluss an der Maschine mit Peristaltik-Pumpe** Der aktuelle Fluss an den Maschinenanschlüssen ergibt sich aus der aktuellen maximalen Einstellung der Maschine über die GUI und die Fußpedalstellung.

Somit gilt für den Fluss an der Maschine:

$$\dot{V}_{\text{Maschine}} = \dot{V}_{\text{GUI}} * \text{FPW} \quad (5.5)$$

und für den Fluss an der Instrumentspitze unter Berücksichtigung des Verstopfungsgrades:

$$\dot{V}_{\text{Instrument}} = \dot{V}_{\text{Maschine}} * (1 - \text{Okklusion}) \quad (5.6)$$

wobei Okklusion den Verstopfungsgrad der Instrumentspitze bezeichnet.

Liegt keine Verstopfung vor, entspricht Okklusion = 0. Ist das Instrument vollständig verstopft, wird Okklusion auf 1 gesetzt. Der Wert ist im Trainingsmodul zu setzen.

**Das Vakuum an der Maschine mit Peristaltik-Pumpe** Das Vakuum, das an der Maschinen-GUI eingestellt wird, stellt eine Obergrenze dar, bis zu der ein Fluss durch die Pumpe erzeugt wird. Das Vakuum an der Instrumentspitze ergibt sich indirekt und ist nur für die Haltekraft der Linsenstücke am Instrument verantwortlich. Es hat keinen direkten Einfluss auf den Aspirationsfluss. Das berechnete Vakuum ist abhängig vom Leitungswiderstand und dem Verstopfungsgrad, das entwickelte Modell wird in Abschnitt 5.4.3.5 behandelt.

### 5.4.3.2 Venturi-Pumpe

Beim Modell der Venturi-Pumpe wird davon ausgegangen, dass das von der Maschine erzeugte Vakuum dem am Maschinenanschluss entspricht. Dieses wird in einen Flusswert am Instrument umgerechnet. Durch die Umrechnung können die später hergeleiteten Berechnungen für das Verhalten der Schläuche weiterverwendet werden.

**Das Vakuum an der Maschine mit Venturi-Pumpe** Das von der Pumpe zu erzeugende Vakuum ist direkt an der Maschinen-GUI einstellbar und wird über das Fußpedal gesteuert. Somit gilt für das Vakuum an der Maschine:

$$p_{\text{Maschine}} = p_{\text{GUI}} * \text{FPW} \quad (5.7)$$

Das Vakuum, das von der Pumpe erzeugt wird, lässt die Flüssigkeit durch den Schlauch fließen. Durch den Schlauchwiderstand reduziert sich das Vakuum während des ungehinderten Flusses. Das dadurch reduzierte Vakuum liegt an der Instrumentspitze an. Dieser Druckunterschied wird genutzt, um den Fluss durch die Instrumentöffnung zu berechnen.

**Der Fluss an der Maschine mit Venturi-Pumpe** Bei einer Venturi-Pumpe wird der Aspirationsfluss indirekt geregelt. Es gibt hierfür keinerlei Einstellungsmöglichkeit an der Maschinen-GUI. Der Fluss an der Instrumentspitze ist abhängig von den geometrischen Eigenschaften.

Abhängig vom Druckunterschied zwischen Maschinenanschluss und Instrument wird das Vakuum in einen Flusswert umgerechnet. Die Berechnungen erfolgen über das Gesetz von Hagen-Poiseuille (Gleichung 5.4) und berücksichtigen die Aspirationsschlauchlänge und den -radius. Die Auswirkung der Verstopfung wird als Reduzierung des Radius berücksichtigt. Für den Fluss an der Instrumentspitze gilt:

$$\dot{V}_{\text{Instrument}} = \frac{\pi(r(1 - \text{Okklusion}))^4}{8\eta l} p_{\text{Maschine}} \quad (5.8)$$

mit den entsprechenden Werten für den Radius  $r$  und die Länge  $l$ .

#### 5.4.3.3 Rise time

Viele Maschinenhersteller bieten die Möglichkeit, die Geschwindigkeit mit der das Vakuum aufgebaut wird, über die rise time zu kontrollieren. Sie entspricht der Zeitspanne, die benötigt wird bis das voreingestellte Vakuumlimit bei vollständig verstopfter Instrumentspitze erreicht wird.

Bei der Peristaltik-Pumpe ergibt sich die rise time implizit aus dem Flusswert und den Schlaucheigenschaften.

Bei der Venturi-Pumpe kann die rise time über die Maschinen-GUI eingestellt werden.

Sie hat Auswirkung auf den Vakuumanstieg während des Fix-Betriebs:

$$p_{\text{Maschine}} = p_{\text{GUI}} * f_{\text{RT}}(t, \text{FPW}) \quad (5.9)$$

wobei  $p_{\text{Maschine}}$  der an der Maschine herrschende Vakuumwert,  $p_{\text{GUI}}$  der über die GUI eingestellte Vakuumwert, FPW der Fußpedalwert und  $f_{\text{RT}}(t, \text{FPW})$  der zeitabhängige rise time Wert ist. Dabei wird der Vakuumwert  $p_{\text{Maschine}}$  wie bei den Maschinen üblich als positiver Wert angegeben. Über die zeitabhängige Funktion  $f_{\text{RT}}(t, \text{FPW})$  kann dieses Verhalten modelliert werden.

Für eine lineare Anstiegszeit gilt:

$$f_{\text{RT}}(t, \text{FPW}) = \begin{cases} 0, & \text{wenn FPW} = 0 \\ \frac{1}{t_{\text{RT}}} * t, & \text{wenn FPW} = 1 \text{ und } t < t_{\text{RT}} \\ 1, & \text{sonst} \end{cases} \quad (5.10)$$

mit der rise time  $t_{\text{RT}}$  und der relativen Zeit  $t$ .

Für die relative Zeit  $t$  gilt:

$$t = \begin{cases} 0, & \text{wenn FPW} = 0 \\ t = t + \Delta t, & \text{wenn FPW} = 1 \end{cases} \quad (5.11)$$

mit dem Zeitschritt  $\Delta t$  der Simulation.

#### 5.4.3.4 Venting

Beim Venting wird Luft oder Flüssigkeit in die Aspirationsleitung eingebracht. Dadurch reduziert sich der Fluss an der Instrumentspitze oder bei einer Verstopfung das Vakuum in der Leitung. Die Maschinenregelung muss dafür Sorge tragen, dass das Venting zum entsprechenden Zeitpunkt aktiviert wird. Dies ist zum Beispiel bei dem Übergang von Fußpedalstellung **B** nach **A** der Fall. Es können je nach Maschine auch andere Mechanismen zum Tragen kommen.

Das Venting stellt eine Fluss- bzw. Vakuumsänderung dar. Dies kann über eine Funktion abgebildet werden, die das Vakuum bzw. den Fluss reduziert. In dem hier entwickelten Modell wird Venting nur durch das Fußpedal gesteuert.

Bei der Peristaltik-Pumpe wird das Venting als Reduzierung des Flusses realisiert. Somit gilt für den Fluss:

$$\dot{V}_{\text{Maschine}} = \dot{V}_{\text{GUI}} * f_{\text{VT}}(t, \text{FPW}) \quad (5.12)$$

Entsprechend wird bei der Venturi-Pumpe das Vakuum durch das Venting reduziert. Somit gilt für das Vakuum:

$$p_{\text{Maschine}} = p_{\text{GUI}} * f_{\text{VT}}(t, \text{FPW}) \quad (5.13)$$

Wobei  $f_{\text{VT}}(t, \text{FPW})$  eine zeitabhängige Funktion ist, für die gilt:

$$f_{\text{VT}}(t, \text{FPW}) = \begin{cases} 1, & \text{wenn FPW in Bereich } \mathbf{B} \text{ und } \mathbf{C} \\ 1 - \frac{t}{t_{\text{VT}}}, & \text{wenn FPW von } \mathbf{B} \text{ nach } \mathbf{A} \text{ wechselt und } t < t_{\text{VT}} \end{cases} \quad (5.14)$$

mit der Zeit  $t_{\text{VT}}$ , die zur vollständigen Reduzierung der Vakuums benötigt wird und der relativen Zeit  $t$  wie sie in 5.11 eingeführt wurde.

#### 5.4.3.5 Einfluss der Schläuche

Zwei Parameter der verwendeten Schläuche haben Einfluss auf das Fluss- und Druckverhalten an der Instrumentspitze. Dies sind die Länge und die Compliance. Die Länge wirkt sich druck- und flussmindernd aus und die Compliance kann einen sehr starken zusätzlichen Fluss auslösen.

Die druckmindernde Wirkung, während des hydrodynamischen Verhaltens, der Schlauchlänge kann auf Reibungsverluste zurückgeführt werden. Allgemein gilt für den Volumenstrom durch Rohre mit kreisrundem Querschnitt das Gesetz von Hagen-Poiseuille (Gleichung 5.4).

Der Druck an einer bestimmten Stelle  $l$  in der Leitung kann damit bei bekanntem Volumenstrom ermittelt werden:

$$p_l = p_0 - \frac{8\eta l}{\pi r^4} \dot{V} \quad (5.15)$$

Der Druck an der Instrumentspitze ist somit von der Schlauchlänge abhängig. Er lässt sich mit Gleichung 5.15 berechnen.

Der zweite Parameter ist die Compliance der Schläuche. Die Compliance beschreibt den Zusammenhang zwischen Druck und Volumen bei umwandeten Strukturen (siehe Abb. 5.10). Die in dieser Arbeit verwendeten Compliance-Funktionen dienen zur Umrechnung von Volumen zu Druck. Sie stellen somit eine Umkehrfunktion des eigentlich als Compliance bekannten Verhältnisses

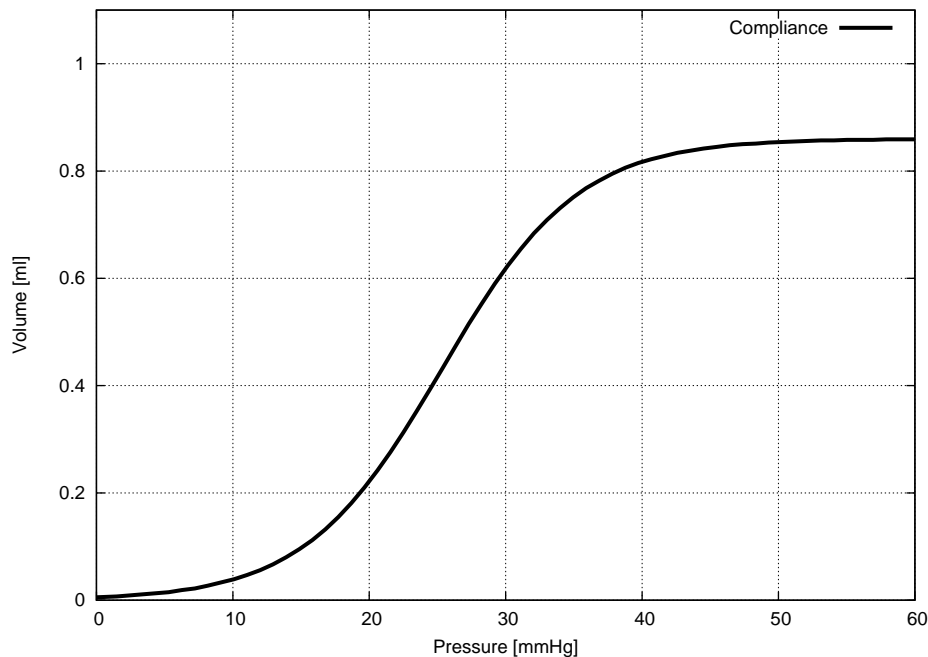


Abbildung 5.10: Die Compliance-Funktion - Der nichtlineare Zusammenhang zwischen Druck und Volumen für eine umwandete Struktur.

von Druck zu Volumen dar. Zur Vereinfachung wird weiterhin der Begriff Compliance-Funktion benutzt.

Die Volumenänderung innerhalb des Schlauches führt zu einer Deformation, dabei wird Energie gespeichert. Solche Volumenänderungen werden durch Verstopfungen an der Instrumentspitze verursacht. Bricht solch eine Verstopfung auf, wird die gespeicherte Energie wieder schlagartig freigesetzt. Dies führt kurzzeitig zu einem hohen Fluss an der Instrumentspitze, der als *Postocclusion-Surge* oder kurz *Surge* bezeichnet wird [80].

**Verhalten des Schlauches während einer Verstopfung** Eine typische Verstopfung läuft in vier Phasen ab. Diese Phasen sind:

1. der freie Fluss,
2. der Beginn der Verstopfung,
3. die vollständige Verstopfung,
4. die Verstopfung bricht auf.

Für eine Modellbildung ist es wichtig, den Ablauf genau zu kennen. Deshalb werden die einzelnen Phasen näher beschrieben.



## 1. Der freie Fluss

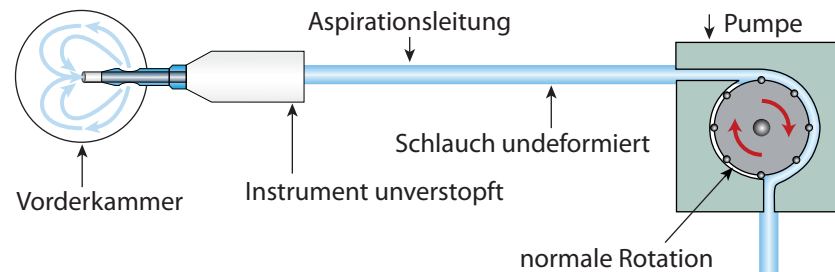


Abbildung 5.11: Das unverstopfte Instrument erzeugt ein Flussfeld in der Vorderkammer bei normal rotierender Pumpe.

Solange die Instrumentspitze nicht verstopft ist (siehe Abb. 5.11), entspricht der Fluss an der Instrumentspitze dem Fluss an der Maschine. Das Vakuum am Sensor entspricht dem Druckunterschied, der aufgebracht werden muss, um den Leitungswiderstand zu überwinden.

## 2. Der Beginn der Verstopfung

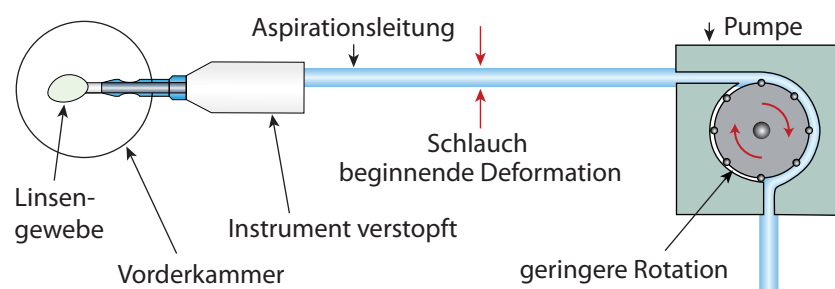


Abbildung 5.12: Zu Beginn der Instrumentverstopfung baut sich ein Vakuum in der Aspirationsleitung auf und die Pumpe rotiert langsamer.

Wird Linsenmaterial vom Instrument angesaugt und verstopft die Spitze, so verringert sich der Fluss an der Spitze. Die Pumpe versucht weiterhin, den Fluss an der Maschine aufrecht zu erhalten. Es wird weiterhin Flüssigkeit aus

dem Aspirationsschlauch evakuiert. Dies erzeugt einen Unterdruck in der Aspirationsleitung und die Aspirationsleitung beginnt sich zu deformieren (siehe Abb. 5.12). Das Vakuum am Sensor setzt sich dabei aus zwei Werten zusammen. Zum einen aus dem Druck, der aufgebracht werden muss, um den Leitungswiderstand zu überwinden und zum anderen aus dem Druck, der durch die Verformung des Schlauches auf die Flüssigkeit ausgeübt wird.

### 3. Die vollständige Verstopfung

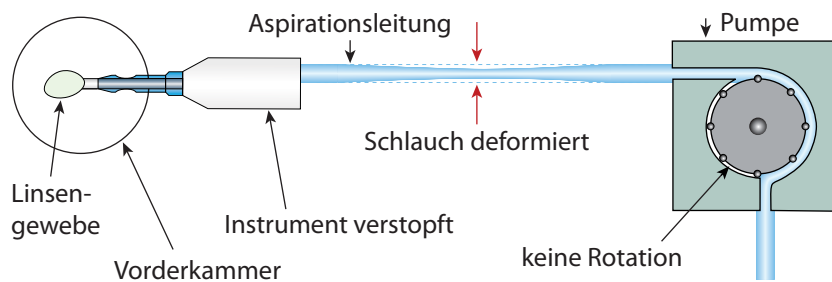


Abbildung 5.13: Die maximale Schlauchdeformation wird bei vollständiger Instrumentverstopfung erreicht. Sie ist abhängig vom an der Maschine eingestellten Vakuumlimit. Die Pumpe rotiert nicht mehr.

In Abbildung 5.13 ist eine vollständige Verstopfung erreicht, der Fluss kommt zum Erliegen. Dies gilt für die Instrumentspitze und die Maschine.

Das Vakuum am Sensor entspricht dem Druck, der durch die Verformung des Schlauches auf die Flüssigkeit ausgeübt wird. Dies entspricht dem Vakuum an der Spitze.

### 4. Die Verstopfung bricht auf

In Abb. 5.14 bricht die Verstopfung auf, Linsengewebe wird durch das Instrument abgesaugt. Der Fluss an der Instrumentspitze und an der Pumpe setzt wieder ein. Er setzt sich aus dem Maschinenfluss und dem Fluss, der durch die Ausdehnung des Schlauches erzeugt wird, zusammen. Das Vakuum im Schlauch reduziert sich auf den Leitungswiderstand. Der Fluss kann zu diesem Zeitpunkt kurzzeitig wesentlich höher sein, als der an der Maschine eingestellte Fluss. Der Surge führt zu Kammerinstabilitäten. Dabei kann es zu Verletzungen des umliegenden Gewebes kommen. Er ist eine Komplikation, die zu vermeiden ist.

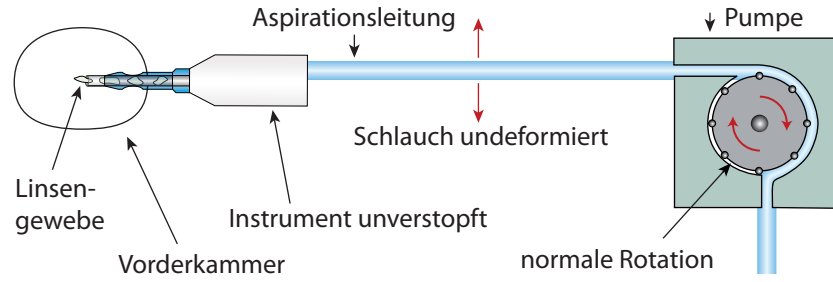


Abbildung 5.14: Die Verstopfung bricht auf. Die Pumpe beginnt zu rotieren. Der Fluss an der Instrumentspitze setzt wieder ein, verstärkt um den Anteil der zurückgehenden Schlauchdeformation. Dies kann zum Surge führen und sich auf die Stabilität der Vorderkammer auswirken.

Das zeitliche Verhalten während einer Instrumentverstopfung erinnert an das Auf- und Entladen eines Kondensators über einen Widerstand. Das Auf- und Entladen folgt unterschiedlichen Kennlinien, die abhängig von Material und Pumpenleistung sind. Das Aufladen ist wesentlich langsamer als das Entladen, das schlagartig verlaufen kann.

Ähnlich dem Auf- und Entladen eines Kondensators, abhängig von seiner Kapazität und des Vorwiderstands, wurde für das Schlauchverhalten ein volumenabhängiges Modell entwickelt. Dieses Modell erlaubt es, unter Vorgabe von Materialeigenschaften, Verstopfungsgrad und Maschineneinstellungen, das Auf- und Entladen des Schlauches zu kontrollieren. Dabei entspricht das Aufladen einer Volumenreduzierung und das Entladen der Normalisierung des Schlauchvolumens.

Die Umrechnung von Volumen zu Druck erfolgt dabei über die Compliance  $C(V)$  (ähnlich Abb. 5.10). Damit ist es innerhalb gewisser Grenzen möglich, für das aktuelle Volumen einen aktuellen Druck zu bestimmen.

Das Vakuum im Schlauch kann somit über das aktuelle Schlauchvolumen, das sich über die Flussdifferenz zwischen der Instrumentspitze und dem Maschinenanschluss ergibt, berechnet werden.

Für das Schlauchvolumen für den nächsten Zeitschritt  $t_{i+1}$  gilt:

$$V_{\text{Schlauch}}(t_{i+1}) = \begin{cases} V_{\text{Schlauch}}(t_i) - V_{\text{diff}}, & \text{wenn } V_{\text{Schlauch}}(t_i) > V_{\text{Schlauch}_{\min}} \\ V_{\text{Schlauch}_{\min}}, & \text{wenn } V_{\text{Schlauch}}(t_i) \leq V_{\text{Schlauch}_{\min}} \end{cases} \quad (5.16)$$

mit

$$V_{\text{diff}} = (\dot{V}_{\text{Instrument}} - (\dot{V}_{\text{Maschine}} - f(p_{\text{Schlauch}}))) * \Delta t * \tau \quad (5.17)$$

und

$$\Delta t = (t_{i+1}) - (t_i) \quad (5.18)$$

Für den aktuellen Druck im Schlauch gilt dann:

$$p_{\text{Schlauch}} = C(V_{\text{Schlauch}}) \quad (5.19)$$

Wobei  $C(V)$  eine geeignete Compliance-Funktion ist, um das Volumen in Druck umzurechnen. Für das Schlauchmodell wird eine lineare Compliance-Funktion verwendet. Das minimale Volumen, das in der Leitung verbleibt, ist abhängig vom Compliance-Koeffizienten und dem zulässigen Vakuumlimit, das an der Maschine eingestellt wurde.

Es gilt:

$$V_{\text{Schlauch}_{\min}} = V_{\text{Schlauch}_{\text{norm}}} (1 - \frac{p_{\text{Maschine}}^{\text{Limit}}}{k}) \quad (5.20)$$

wobei  $k$  den materialabhängigen Compliance-Koeffizienten des Schlauches und  $p_{\text{Maschine}}^{\text{Limit}}$  das an der Maschine eingestellte Vakuumlimit bezeichnet.

Über die Zeitkonstante  $\tau$  lässt sich die Dauer regulieren, bis die Leitung auf ihr zulässiges Minimalvolumen evakuiert ist und sich wieder vollständig ausgeglichen hat.

Die Funktion  $f(p_{\text{Schlauch}})$  basiert auf dem Gesetz von Hagen-Poiseuille (Gleichung 5.4) und liefert einen Flusswert am Maschinenanschluss für den anliegenden Druck, der sich aus der Compliance des Schlauches ergibt. Durch diese Rückkopplung wirkt sich der aufgebaute Druck auf die Flussdifferenz aus, bis sich ein Gleichgewicht einstellt.

Solange die Flussdifferenz kleiner 0 ist, baut sich das Vakuum im Schlauch auf. Ist die Flussdifferenz größer 0, bedeutet dies, dass eine Verstopfung aufgebrochen ist und es zum Surge kommen kann. Der Surge-Fluss  $\dot{V}_{\text{Surge}}$  wird über das Gesetz von Hagen-Poiseuille (Gleichung 5.4) unter Berücksichtigung der Instrumentverstopfung berechnet:

$$\dot{V}_{\text{Surge}} = \frac{\pi(r(1 - \text{Okklusion}))^4}{8\eta l} p_{\text{Schlauch}} \quad (5.21)$$

und dem entsprechenden Radius  $r$  und der entsprechenden Länge  $l$ . Der Gesamtfluss an der Instruementspitze ergibt sich aus der Summe des Surge-Fluss und dem Fluss an der Instrumentspitze. Der zusätzliche Surge-Fluss fließt dann bis sich wieder ein Gleichgewicht im Schlauch eingestellt hat.

#### 5.4.3.6 Der Einfluss der Instrumentform und Öffnungen

Die Geometrie des Instruments hat ebenfalls Auswirkungen auf das Fluss- und Vakuumverhalten. Sie wirkt sich nur druckmindernd in Bezug auf den Druckabfall in der Vorderkammer aus, denn das Instrument besitzt keine Compliance. Die Aspirationsöffnung ist die Stelle mit dem geringsten Durchmesser. Hier wird der größte Widerstand auf die strömende Flüssigkeit ausgeübt. Diese Druckverminderung wird ebenfalls durch das Gesetz von Hagen-Poiseuille 5.4 berechnet. Durch Reihenschaltung der einzelnen druckmindernden Anteile fließt die Instrumentform in das Modell ein.

#### 5.4.4 Instrumentspezifische Eigenschaften

Als instrumentspezifische Eigenschaften wurden der Ultraschall beim Phakohandstück und die Schnittrate beim Vitrektor modelliert.

##### 5.4.4.1 Das Phakohandstück

Moderne Operationsmaschinen<sup>2</sup> bieten verschiedene Einstellungsmöglichkeiten für die Ultraschallerzeugung an. Dabei lässt sich an der Maschine die Auslenkung der Nadel und die Ultraschallmodulation einstellen. Die Ultraschallfrequenz ist maschinenspezifisch und kann nicht geändert werden.

Über die Ultraschallmodulation kann die Dauer des Ultraschallimpulses eingestellt werden, nach [80] sind die gebräuchlichsten Ultraschallmodulationen (engl. *ultrasound power modulation*):

- Kontinuierlich,
- Pulse Mode,
- Burst Mode,
- Hyperpulse.

Die verschiedenen Ultraschallmodulationen wurden entwickelt, um möglichst wenig Ultraschall zu benutzen, damit das umliegende Gewebe nicht in Mitleidenschaft gezogen wird.

Die Ultraschallerzeugung wird nicht von der Umgebung beeinflusst. Es besteht keine Rückkopplung, die in der Maschinenregelung verarbeitet werden

---

<sup>2</sup> Der Autor bezieht sich hierbei nur auf Maschinen mit longitudinaler Auslenkung, jüngere Entwicklungen wie *Torsional Phaco* werden nicht berücksichtigt.

müsste. Die an der Maschine eingestellten Werte können deshalb direkt umgesetzt werden und für weitere Berechnungen in der virtuellen Realität genutzt werden.

Da sich die Mediziner und Maschinenhersteller noch im unklaren über Auswirkungen des Ultraschalls (siehe Abschnitt 5.1) auf das Verflüssigen der Linse sind, wird hier eine deskriptive Modellierung vorgeschlagen.

Dafür wird eine quantitative *Ultraschallenergie* pro Simulationsschritt berechnet, die von der Phakonadel erzeugt wird. Diese setzt sich aus der Auslenkung und der Ultraschallmodulation zusammen. In der virtuellen Realität wird dieser Wert verwendet, um das Gewebeverhalten zu beeinflussen.

Für die Ultraschallenergie gilt:

$$E_{\text{USMaschine}} = E_{\text{USGUI}} * \text{FPW} \quad (5.22)$$

Der Fußpedalwert FPW ergibt sich entsprechend für die „lineare“ oder „fixe“ Einstellung wie in Abschnitt 5.4.1.

Der Wert an der Instrumentspitze wird aus dem Maschinenwert und dem gewählten Modus berechnet:

$$E_{\text{USInstrument}} = E_{\text{USMaschine}} * \delta(\text{Modus}) \quad (5.23)$$

Über den Modus wird festgelegt, ob für das aktuelle Frame der Ultraschall aktiv ist, dabei gilt:

$$\delta(\text{Modus}) = \begin{cases} 1, & \text{wenn } t_{\text{aktiv}} < T_{\text{aktiv}}(\text{Modus}) \\ 0, & \text{sonst} \end{cases} \quad (5.24)$$

mit  $t_{\text{aktiv}}$  für die aktuelle Ultraschallzeit. Die verwendeten Aktiv- und Inaktivzeiten  $T_{\text{aktiv}}(\text{Modus})$  und  $T_{\text{inaktiv}}(\text{Modus})$  sind modusabhängige Konstanten.

Die aktuelle Ultraschallzeit berechnet sich folgendermaßen:

$$t_{\text{aktiv}} = \begin{cases} t_{\text{aktiv}} + \Delta t, & \text{wenn } t_{\text{aktiv}} < T_{\text{aktiv}}(\text{Modus}) \\ t_{\text{aktiv}}, & \text{wenn } t_{\text{aktiv}} > T_{\text{aktiv}}(\text{Modus}) \text{ und} \\ & t_{\text{inaktiv}} < T_{\text{inaktiv}}(\text{Modus}) \\ 0, & \text{wenn } t_{\text{aktiv}} > T_{\text{aktiv}}(\text{Modus}) \text{ und} \\ & t_{\text{inaktiv}} > T_{\text{inaktiv}}(\text{Modus}) \end{cases} \quad (5.25)$$

dabei ist  $\Delta t$ , die durch die Simulation vorgegebene Framezeit und  $t_{\text{inaktiv}}$  die aktuelle Inaktivzeit. Für die Inaktivzeit gilt:

$$t_{\text{inaktiv}} = \begin{cases} t_{\text{inaktiv}} + \Delta t, & \text{wenn } t_{\text{inaktiv}} < T_{\text{inaktiv}}(\text{Modus}) \text{ und} \\ & t_{\text{aktiv}} > T_{\text{aktiv}}(\text{Modus}) \\ 0, & \text{wenn } t_{\text{inaktiv}} > T_{\text{inaktiv}}(\text{Modus}) \text{ und} \\ & t_{\text{aktiv}} > T_{\text{aktiv}}(\text{Modus}) \end{cases} \quad (5.26)$$

Die Dauer des Ultraschallimpulses lässt sich über die Aktiv- bzw. Inaktivzeiten der Ultraschallmodulation in den Gleichungen 5.25 und 5.26 einstellen.

Die Aktiv- und Inaktivzeiten wurden nach den Angaben in [80] gewählt. Als Zyklusdauer (die Summe aus Aktiv- und Inaktivzeit) wird im Folgenden eine Sekunde angenommen, zur Vereinfachung wurde auf die Einheit verzichtet.

Für den **Kontinuierlichen** Ultraschallmodus gilt, dass der Ultraschall aktiviert wird, wenn sich das Fußpedal in Position **C** befindet. Dies wird durch folgende Einstellung erreicht:

$$T_{\text{aktiv}}(\text{Kontinuierlich}) = 1 \quad (5.27)$$

$$T_{\text{inaktiv}}(\text{Kontinuierlich}) = 0 \quad (5.28)$$

Beim **Pulse Mode** beträgt die Ultraschallaktivzeit 50% des Kontinuierlichen Ultraschallmodus. Über die „Pulserate“ kann an der Maschine die Anzahl der Aktivierungen eingestellt werden. Die Pulsebreite des einzelnen Pulses reduziert sich dabei. Die Gesamtaktivzeit bleibt erhalten.

Für den Pulse Mode gilt:

$$T_{\text{aktiv}}(\text{Pulse Mode}) = \frac{0,5}{\text{Pulserate}} \quad (5.29)$$

$$T_{\text{inaktiv}}(\text{Pulse Mode}) = \frac{1 - T_{\text{aktiv}}(\text{Pulse Mode})}{\text{Pulserate}} \quad (5.30)$$

Beim **Burst Mode** bleibt die Pulsbreite gleich. In Abhängigkeit von der Fußpedalposition nimmt die Anzahl der Pulse zu. Dies wird durch folgende Aktiv- und Inaktivzeiten modelliert:

$$T_{\text{aktiv}}(\text{Burst Mode}) = 0,125 \quad (5.31)$$

$$T_{\text{inaktiv}}(\text{Burst Mode}) = \frac{1 - T_{\text{aktiv}}(\text{Burst Mode}) * \text{FPW}'}{\text{FPW}'} \quad (5.32)$$

Wobei hier für den Fußpedalwert (FPW') in Abhängigkeit von der Fußpedalposition gilt:

$$\text{FPW}' = \begin{cases} 1, & \text{wenn Fußpedalposition} > 0 \text{ und } \leq 0,125 \\ 2, & \text{wenn Fußpedalposition} > 0,125 \text{ und } \leq 0,25 \\ 3, & \text{wenn Fußpedalposition} > 0,25 \text{ und } \leq 0,375 \\ 4, & \text{wenn Fußpedalposition} > 0,375 \text{ und } \leq 0,5 \\ 5, & \text{wenn Fußpedalposition} > 0,5 \text{ und } \leq 0,625 \\ 6, & \text{wenn Fußpedalposition} > 0,75 \text{ und } \leq 0,825 \\ 7, & \text{wenn Fußpedalposition} > 0,825 \text{ und } \leq 0,95 \\ 8, & \text{wenn Fußpedalposition} > 0,95 \text{ und } \leq 1 \end{cases} \quad (5.33)$$

Der **Hyperpulse Mode** ähnelt dem Pulse Mode, aber er besitzt nur die halbe Pulsbreite.

Es gilt:

$$T_{\text{aktiv}}(\text{Hyperpulse Mode}) = \frac{0,025}{\text{Pulsrate}} \quad (5.34)$$

$$T_{\text{inaktiv}}(\text{Hyperpulse Mode}) = \frac{1 - T_{\text{aktiv}}(\text{Hyperpulse Mode})}{\text{Pulserate}}. \quad (5.35)$$

Für weitere Modi können die Aktiv- und Inaktivzeiten angepasst werden. In der Simulation kann somit der aktuelle Wert für die Zeitspanne zwischen zwei Frames abgefragt werden. Durch Aufsummieren erhält man die gesamte Ultraschalldauer. Dies geschieht in der Maschine für die effektive Phakozeit (engl. *effective phaco time*, kurz EPT) und für die durchschnittliche Ultraschallenergie (engl. *average phaco power*, kurz AVG).

#### 5.4.4.2 Vitrektor

Es werden auch bei anderen Operationen am Auge Instrumente verwendet, deren aktive Eigenschaften über die Maschine steuerbar sind. Die bei der Simulation der Phakoemulsifikation gewonnenen Erkenntnisse werden auf diese übertragen. Solch ein Instrument ist der Vitrektor. Dabei handelt es sich um ein kombiniertes Saug- und Schneideinstrument, das meist bei Operationen am hinteren Augenabschnitt, nahe der Retina, eingesetzt wird. Über die Maschine lässt sich die Schnittrate des Vitrektors steuern. Dies geschieht, wie



beim Ultraschall, Linear oder Fix, bis zum eingestellten Limit, über das Fußpedal. Dabei ist zu beachten, ob es sich um einen symmetrischen oder asymmetrischen Schneidezyklus handelt. Beim symmetrischen Schneidezyklus sind die Dauer des Schließens und Öffnens, sowie die Verweildauer im geschlossenen und offenen Zustand gleich lang. Beim asymmetrischen Schneidezyklus können sich diese Zeiten unterscheiden. Die Erhöhung der Schnittrate wirkt sich zusätzlich flussmindernd aus [40].

Im derzeitigen Modell wird von einem symmetrischen Schneidezyklus ausgegangen. Es gilt somit:

$$\text{Schnittrate}_{\text{Instrument}} = \text{Schnittrate}_{\text{GUI}} * \text{FPW} \quad (5.36)$$

Für den Fußpedalwert FPW gelten die üblichen Annahmen für Linear- und Fix-Betrieb.

Für die Reduzierung des Aspirationsflusses wird ein linearer Zusammenhang angenommen:

$$\dot{V}_{\text{Instrument}} = \dot{V}_{\text{Instrument}} - f(\text{Schnittrate}_{\text{Instrument}}) \quad (5.37)$$

mit

$$f(\text{Schnittrate}_{\text{Instrument}}) = \text{Schnittrate}_{\text{Instrument}} * \text{CFR} \quad (5.38)$$

Die CFR (engl. *Cutrate Flow Reduction*) ist frei wählbar und muss auf den jeweiligen Vitrektor angepasst werden, im aktuellen Modell wurde CFR = 0.01 gewählt.

Der so ermittelte Fluss dient als Grundlage für die Volumen- und Druckverhältnisse im Auge, die im *Kapitel 6 - Gewebeinteraktion mit dem Instrumentarium* behandelt werden.

### 5.4.5 Ablauf eines Simulationsschrittes

Der Simulationsschritt läuft sequentiell ab und berechnet die Auswirkungen des jeweiligen Modells auf das gesamte Verhalten der OMS. Es wird zwischen der Simulation für die Phakoemulsifikation (siehe Algorithmus 1) und für die Vitrektomie (siehe Algorithmus 2) unterschieden.

Die Berechnungen in den einzelnen Schritten beziehen sich auf die oben entwickelten Modelle. Der Irrigationsdruck an der Instrumentspitze wird durch

---

**Algorithmus 1** Der Ablauf der Maschinen-Simulation für die Phakoemulsifikation

---

```
1: while Simulation aktiv do
2:   berechne den Irrigationsdruck für die aktuelle Flaschenhöhe
3:   if Peristaltik-Pumpe then
4:     berechne den Fluss an der Maschine unter Berücksichtigung des Ventings
5:     berechne den Fluss am Instrument abhängig vom Maschinenfluss und der Instrumentverstopfung
6:   end if
7:   if Venturi-Pumpe then
8:     berechne das Vakuum an der Maschine unter Berücksichtigung der rise time und des Ventings
9:     berechne den Fluss am Instrument abhängig vom Maschinenvakuum und der Instrumentverstopfung
10:  end if
11:  berechne die Auswirkungen des Aspirationsschlauches auf Fluss und Vakuum an der Instrumentspitze
12:  berechne die Ultraschallenergie
13: end while
```

---

---

**Algorithmus 2** Der Ablauf der Maschinen-Simulation während der Vitrektomie

---

```
1: while Simulation aktiv do
2:   berechne den Irrigationsdruck für die aktuelle Flaschenhöhe
3:   if Peristaltik-Pumpe then
4:     berechne den Fluss an der Maschine unter Berücksichtigung des Ventings
5:     berechne den Fluss am Instrument abhängig vom Maschinenfluss und der Schnittrate
6:   end if
7:   if Venturi-Pumpe then
8:     berechne das Vakuum an der Maschine unter Berücksichtigung der rise time und des Ventings
9:     berechne den Fluss am Instrument abhängig vom Maschinenvakuum und der Schnittrate
10:  end if
11: end while
```

---

die aktuelle Flaschenhöhe und die Stellung des Ventils bestimmt. Abhängig vom Pumpentyp werden die Werte für den Fluss und das Vakuum an der Instrumentspitze berechnet. Die endgültigen Werte werden durch den Einfluss des Schlauchsystems und des Rückkopplungswertes für Verstopfung berechnet. Die Berechnung der Ultraschallenergie liefert die fiktive Ultraschallenergie für den aktuellen Modus und die Fußpedalstellung.

Der Ablauf der Maschinen-Simulation während der Vitrektomie (siehe Algorithmus 2) unterscheidet sich von dem Ablauf für die Phakoemulsifikation. Der Fluss am Instrument wird nur durch die Schnittrate reduziert. Die Auswirkungen der Schläuche werden nicht ausgewertet, da es in der Realität keinen Surge gibt. Die Schnittrate wird über die GUI und das Fußpedal gesteuert.

Die während der OMS berechneten Werte werden als Ist-Wert für die Simulation abgespeichert und können in der virtuellen Umgebung verwendet werden.

## 5.5 Ergebnisse

Die vorgestellten Ergebnisse für die virtuelle Maschine wurden mit dem *Maschinentester* ermittelt. Der Maschinentester ist eine Testumgebung, um die Abläufe in der Maschine zu testen und zu visualisieren. Er wird genauer im *Kapitel 7 - Software-Architektur* vorgestellt. Die hier aufgeführten Messwerte und Grafiken decken nur einen Teil der mit dem Maschinentester möglichen Testszenarien ab.

Als Vergleichswerte dienen Messungen, die an einer Geuder Megatron S3 mit Peristaltik-Pumpe, Phakohandgriff, aufgeschraubter Phakonadel, Irrigationsaufsatz und Schlauchsystem durchgeführt wurden. In Tabelle 5.2 sind die Abmessungen der verwendeten Schläuche aufgeführt.

	Aspiration	Irrigation
Länge	2 m	2,3 m
Außendurchmesser	4 mm	5 mm
Innendurchmesser	1,6 mm	3,4 mm

Tabelle 5.2: Die Abmessungen des verwendeten Schlauchsystems bei der Geuder Megatron S3.

Die Phakospitze besitzt folgenden Abmessungen: 21 mm Länge und 0,6 mm

Innendurchmesser. Darüber ist der Irrigationsaufsatz gestülpt. Dieser besitzt zwei Öffnungen an der Instrumentspitze.

### 5.5.1 Irrigation

Flaschenhöhe [cm]	Geuder Fluss [ml/min]	OMS Fluss [ml/min]
30	35	29
40	45	39
60	65	59
80	76	79
100	87	98

Tabelle 5.3: Das Verhältnis von Fluss zu Flaschenhöhe gemessen an der Geuder Megatron S3 mit einem Phakohandgriff, aufgeschraubter Phakospitze, Irrigationsaufsatz, Schlauchsystem und der OMS.

**Flaschenhöhe und Fluss** In Tabelle 5.3 ist der ungehinderte Fluss an der Geuder Megatron S3 mit einem Phakohandgriff, aufgeschraubter Phakospitze, Irrigationsaufsatz, Schlauchsystem und der OMS für verschiedene Flaschenhöhen dargestellt. In der Simulation werden für die Irrigationsleitung die gleichen Abmessungen verwendet wie bei der Geuder Megatron S3. Für den Irrigationsaufsatz und das Handstück wird ein Ersatzwiderstandswert verwendet, der sich aus der Länge des Irrigationsaufsatzes von 19 mm und einem angenommenen Radius von 0,3 mm ergibt. Dieser Ersatzwiderstand wurde aufgrund der unbekannten und nicht eindeutigen Geometrie des Handsstücks gewählt. Für die berechneten Werte gilt hier ein linearer Zusammenhang. Die gemessenen Werte weichen davon ab, was auf die Objektgeometrie und die Deformierbarkeit der realen Schläuche zurückgeführt werden kann. Schon kleinste Änderungen in der realen Objektgeometrie führen zu veränderten Flusswerten.

Bei dieser Messreihe handelt es sich um hypothetische maximale Flusswerte, die während der Operation kaum erreicht werden, da durch die Vorderkammer ein Gegendruck erzeugt wird. Die Auswirkungen der Vorderkammer auf das Irrigationsverhalten wird in Abschnitt 6.2.1 genauer untersucht. Es hat sich gezeigt, dass die Abweichungen des ungehinderten Flusses ignoriert werden können, solange sich das Gesamtsystem plausibel verhält.

**Flusswiderstand der Schläuche** Neben dem freien Fluss in die Umgebung gibt Seibel [80] eine Messreihe zum Durchfluss durch das gesamte Schlauchsystem an. Diese Messreihe gibt Informationen für den maximal möglichen Fluss mit dem verwendeten Schlauchsystem bei einer bestimmten Flaschenhöhe an. Werden höhere Aspirationsflusswerte durch die Maschine erzeugt, als durch die Irrigation bereitgestellt werden können, führt dies zu Kammerinstabilitäten. In Tabelle 5.4 sind die Werte aus [80] dargestellt. In Tabelle 5.5 die durch die OMS berechneten Werte. Über die VRmagic Maschinen-GUI wird die Flaschenhöhe in cm eingestellt. Dadurch ergeben sich leichte Abweichungen der Grundeinstellungen.

Flaschenhöhe [inch]	20	25	30	35	40
Flaschenhöhe [cm]	50,8	63,5	76,2	88,9	101,6
Fluss [ml/min]	14	16	18	20	22

Tabelle 5.4: Das Verhältnis von Flaschenhöhe zu Durchfluss nach Seibel [80].

Flaschenhöhe [inch]	19,7	23,6	29,5	35,4	39,4
Flaschenhöhe [cm]	50	60	75	90	100
Fluss [ml/min]	14	17	21	26	28

Tabelle 5.5: Das Verhältnis von Flaschenhöhe zu Durchfluss berechnet mit der OMS.

Im Vergleich zum ungehinderten Fluss erkennt man deutlich eine Flussreduzierung. Dies ist auf den höheren Schlauchwiderstand, der durch die Verlängerung um den dünneren Aspirationsschlauch entsteht, zurückzuführen. Die mit der OMS berechneten Werte sind geringfügig höher als die gemessenen, dies führt zu einem gutmütigeren Verhalten während der Simulation. Gutmütig bedeutet in diesem Fall, es kommt nicht so schnell zu Kammerinstabilitäten. Die Abweichungen erlauben eine plausible Simulation des Druck- und Volumenverhaltens der Vorderkammer (siehe Abschnitt 6.7).

### 5.5.2 Aspiration

**Verhältnis Fluss zu Vakuum** In Tabelle 5.6 ist das Verhältnis Fluss zu Vakuum dargestellt. Die an der Geuder Megatron S3 gemessenen Werte be-

ziehen sich auf die oben aufgeführte Konfiguration der Maschine. Es wurde eine Messung mit einem Standard-Tip (Außendurchmesser: 1,17 mm) und einem Micro-Tip (Außendurchmesser: 0,8 mm) durchgeführt. Die Ansteuerung der Rollenpumpen erfolgte im Fix-Betrieb, das Vakuumlimit wurde auf 600 mmHg eingestellt, so dass die Maschine nicht frühzeitig abregelt. Die Geuder Megatron S3 liefert den Vakuumwert immer in Fünferschritten. Dabei kam es vor, dass der Wert für einen Flusswert schwankte, dies ist durch den erweiterten Wertebereich gekennzeichnet. Betrachtet man die Messwerte, wird deutlich, wie stark die Geometrie der Phakonadel den Vakuumwert am Maschinensensor beeinflusst. Für das Standard-Tip sind die berechneten Werte im Vergleich zu den gemessenen Werten höher. Dies liegt an der für die Simulation vereinfachten Objektgeometrie. Beim Micro-Tip ergibt sich eine komplexere Situation. Für geringe Flusswerte liegen die berechneten Vakuumwerte über den gemessenen Werten. Mit steigenden Flusswerten kehrt sich das Verhältnis um und die gemessenen übersteigen die berechneten Werte. Dies lässt sich auf die Vereinfachungen wie laminare Strömungen und kreisrunde Leitungsquerschnitte, die für die OMS angenommen wurden, zurückführen.

Die Abweichungen können als gering angesehen werden. Ein Einsatz scheint zurzeit bedenkenlos.

Fluss [ml/min]	10	15	20	25	30	35	40	45	50
Geuder Schlauch	15	20	35	50	60	70	85		
Vakuum [mmHg]	-	-	-	-	-	-	-		
Geuder Standard-Tip	20	25	40	55	65	75	90	100	110
Vakuum [mmHg]	-	-	-	-	-	-	-	-	-
Geuder Micro-Tip	20	35	50		85	110	130	155	
Vakuum [mmHg]	25	40	55	70	90	115	135	160	180
OMS Standard-Tip		125							
Vakuum [mmHg]	70	130	200	290	365	460	535	540	545
OMS Micro-Tip									
Vakuum [mmHg]	45	67	90	112	134	157	180	202	224
OMS Standard-Tip									
Vakuum [mmHg]	91	137	182	228	274	319	365	411	456

Tabelle 5.6: Das Verhältnis Fluss zu Vakuum, gemessen an der Geuder Megatron S3 mit Peristaltik-Pumpe und der simulierten OMS, unverstopft.

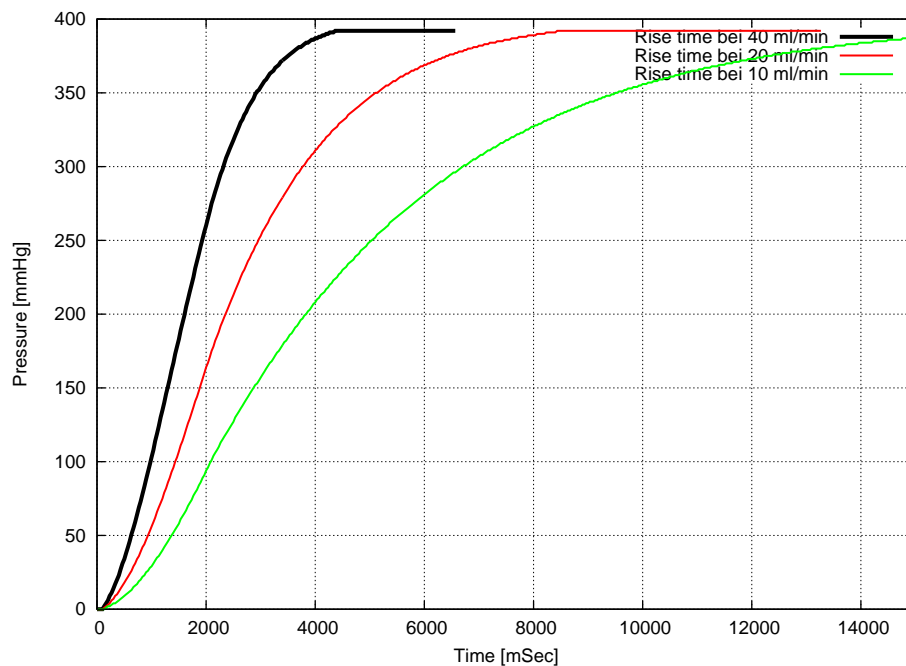


Abbildung 5.15: Die rise time der OMS mit Peristaltik-Pumpe.

**Rise time** In [80] wird ein linearer Zusammenhang für die rise time angegeben. In Abbildung 5.15 sind die simulierten Werte aufgetragen. Bei der OMS ergibt sich für die Peristaltik-Pumpe der errechnete Wert für das Vakuum am Maschinensensor über die Schlauch-Compliance. Dies spiegelt sich in dem nichtlinearen Zusammenhang wider. Es lässt sich deutlich erkennen, dass sich die Flussstärke auf den Vakuumanstieg auswirkt. In Abbildung 5.16 ist die rise time für die simulierte Venturi-Pumpe dargestellt. Hierbei gilt ein linearer Zusammenhang, der sich über die rise time variieren lässt. Für die Messungen wurde die Instrumentverstopfung auf 100% eingestellt und die Pumpen im Fix-Betrieb verwendet.

**Auf- und Entlade-Vorgänge des Aspirationsschlauches** Da es sich hier um das erste veröffentlichte Modell handelt, das die Compliance des Aspirationsschlauches berücksichtigt, kann es nicht mit anderen verglichen werden. Um die Auswirkungen der Parameter auf das Schlauchmodell dennoch beurteilen zu können, wurden mehrere Messungen durchgeführt. In den Abbildungen 5.17, 5.18 und 5.19 ist das Auf- und Entladeverhalten des Aspirationsschlauches für verschiedene Steifigkeitswerte  $k$  und Zeitkonstanten  $\tau$  für die Ladewerte dargestellt.

Über die Schlauchsteifigkeit lässt sich das minimal im Schlauch verbleibende

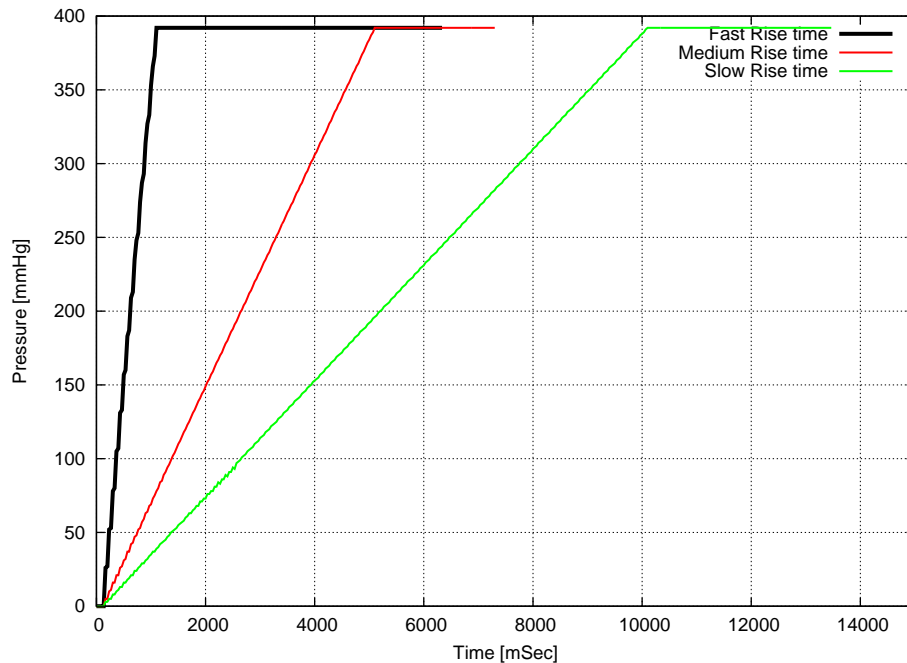


Abbildung 5.16: Die rise time der OMS mit Venturi-Pumpe.

Volumen bestimmen. Je geringer dieses ist, desto stärker ist der dadurch erzeugte Surge in der Vorderkammer. Über die  $\tau$ -Werte lassen sich die zeitlichen Auswirkungen kontrollieren. Dies ist die Aufladedauer bis das minimale Volumen erreicht ist und die Entladedauer bis sich der Schlauchzustand wieder normalisiert hat.

Für das Auf- und Entladeverhalten wurden jeweils exemplarisch drei  $\tau$ -Werte verwendet. Dies sind für das Aufladen  $\tau_a = 2$ ,  $\tau_a = 5$  und  $\tau_a = 10$  und für das Entladen  $\tau_e = 2$ ,  $\tau_e = 20$  und  $\tau_e = 50$ . Je größer der  $\tau$ -Wert, desto schneller ist das Schlauchvolumen wieder ausgeglichen. Beeinflusst wird dieser Wert zusätzlich von der Schlauchsteifigkeit und den Ladewerten. Denn für hohe Steifigkeit und geringe Ladewerte wird in derselben Ladezeit ein nicht so geringeres Schlauchvolumen erzeugt. Der Ausgleich erfolgt somit schneller. Der dadurch erzeugte Surge-Flow ist ebenfalls geringer.

In Abbildung 5.17 ist ein in der Realität nicht vorkommendes Verhalten zu erkennen. Das über die Schlauchfestigkeit und das Maschinenvakuumlimit erzeugte minimale Volumen ist dabei größer als das Schlauchvolumen. Dadurch kann es bei hohen  $\tau$ -Werten für das Ladeverhalten zu einem negativen Schlauchvolumen kommen. Der Einsatz einer geringen Schlauchfestigkeit führt zu einem langsamen Anstieg des Schlauchvakuums. Dies erzeugt ein schwächeres Surge-Verhalten. Somit wäre eine Verwendung dieser Parameter für Anfänger dennoch geeignet. Das negative Schlauchvolumen muss in der Simulation vermieden werden. Dies erfolgt durch Begrenzung der Wer-



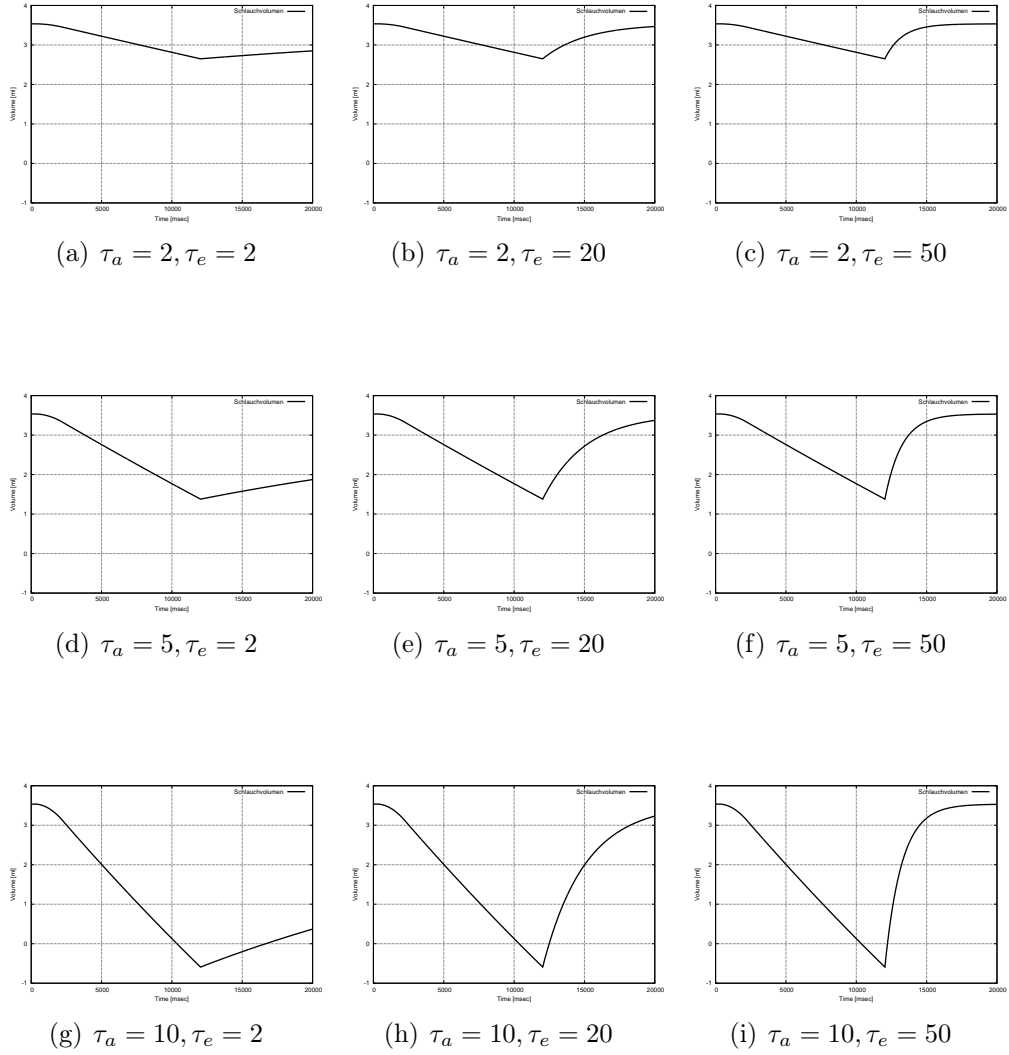
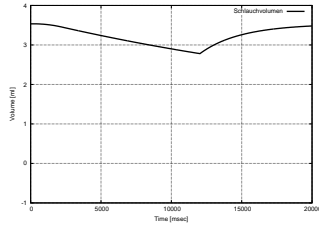
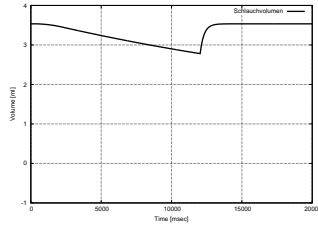


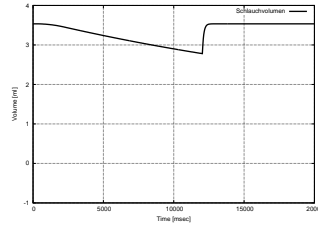
Abbildung 5.17: Das zeitabhängige Schlauchvolumen für einen Compliance-Koeffizienten von  $k = 50$  mmHg und verschiedenen Zeitkonstanten  $\tau_a$  und  $\tau_e$  für das Auf- und Entladen.



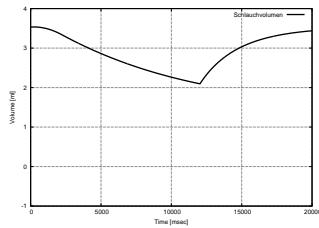
(a)  $\tau_a = 2, \tau_e = 2$



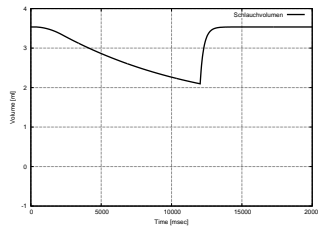
(b)  $\tau_a = 2, \tau_e = 20$



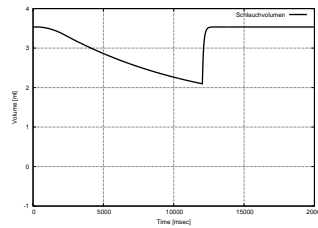
(c)  $\tau_a = 2, \tau_e = 50$



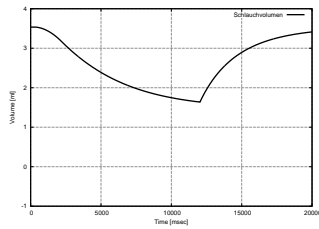
(d)  $\tau_a = 5, \tau_e = 2$



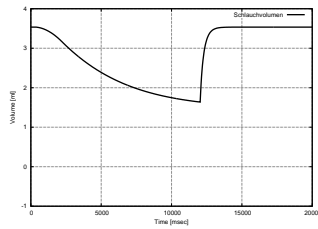
(e)  $\tau_a = 5, \tau_e = 20$



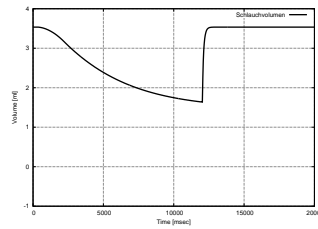
(f)  $\tau_a = 5, \tau_e = 50$



(g)  $\tau_a = 10, \tau_e = 2$



(h)  $\tau_a = 10, \tau_e = 20$



(i)  $\tau_a = 10, \tau_e = 50$

Abbildung 5.18: Das zeitabhängige Schlauchvolumen für einen Compliance-Koeffizienten von  $k = 500$  mmHg und verschiedenen Zeitkonstanten  $\tau_a$  und  $\tau_e$  für das Auf- und Entladen.

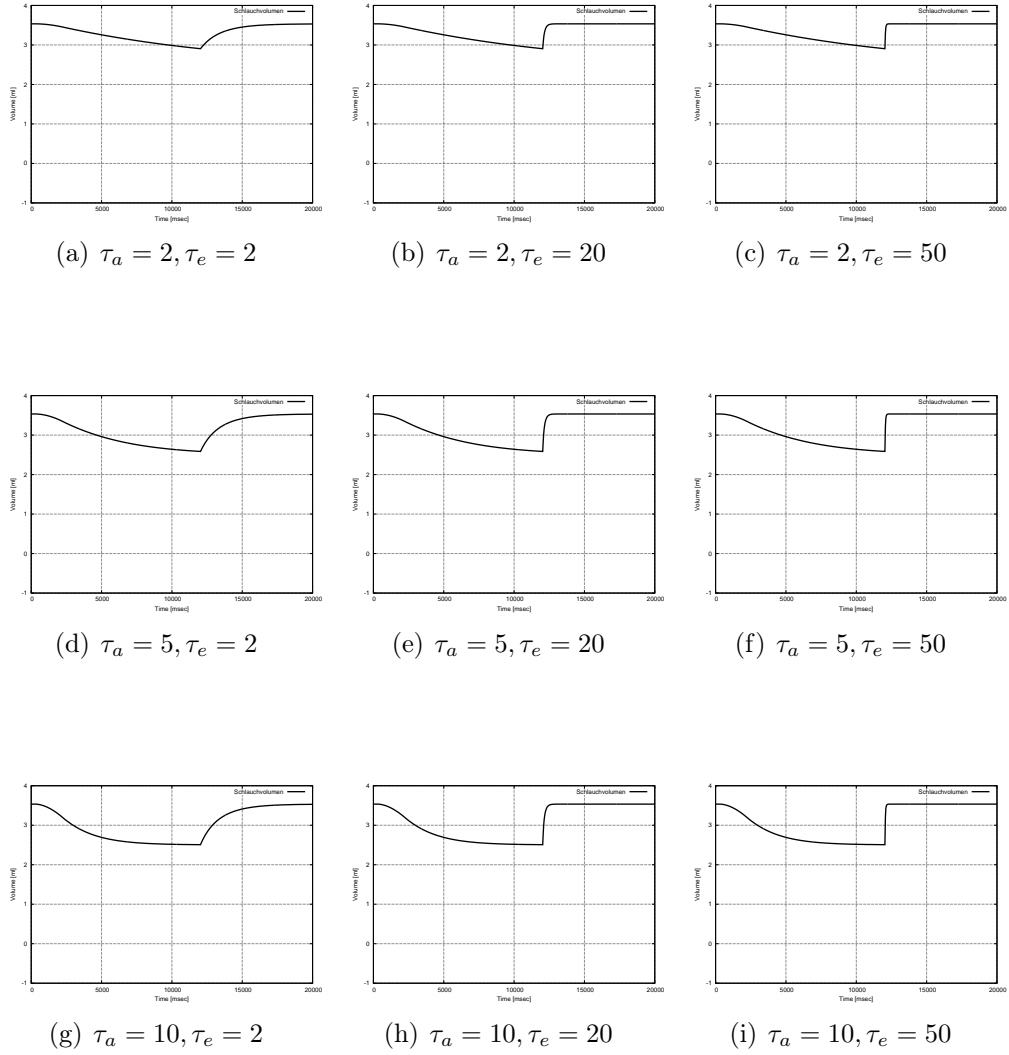


Abbildung 5.19: Das zeitabhängige Schlauchvolumen für einen Compliance-Koeffizienten von  $k = 1000$  mmHg und verschiedenen Zeitkonstanten  $\tau_a$  und  $\tau_e$  für das Auf- und Entladen.

te auf ein Minimum bzw. ein Maximum. In der Realität ist dies über die Materialeigenschaften vorgegeben.

Durch die Wahl der Schlauchsteifigkeit sowie der Lade- und Entladewerte lässt sich die Stärke des Surge-Effekts einstellen. Für die aktuelle Maschinen-Simulation wurden folgende Parameter gewählt:  $k = 1000$  mmHg,  $\tau_a = 2$  und  $\tau_e = 50$ . Dies erzeugt im Zusammenspiel mit der Simulation der Vorderkammer moderates Surge-Verhalten.

### 5.5.3 Ultraschallmodulation

Die in der OMS verwendeten Ultraschallmodulationen werden hier mit den Werten aus der Theorie [80] verglichen. Die Tests orientieren sich streng an den Vorgaben in [80]. Der Maschinentester wurde jeweils entsprechend konfiguriert.

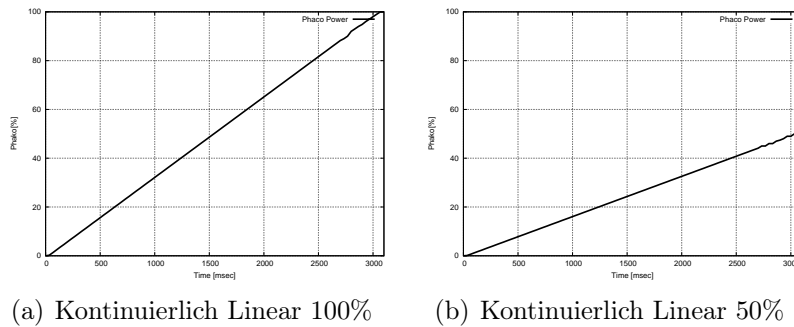


Abbildung 5.20: Die kontinuierlich lineare Ultraschallmodulation mit unterschiedlichen maximal Werten.

**Kontinuierlich Linear** Der Fußpedalwert wurde über drei Sekunden kontinuierlich erhöht, bis er den Vollausschlag erreichte. Für die erste Messung wurde das Maximum auf 100% eingestellt. Nach Ablauf der drei Sekunden war das Maximum erreicht. Der Verlauf ist in Abbildung 5.20(a) dargestellt. Für die zweite Messung wurde das Maximum auf 50% reduziert. Der Verlauf ist in Abbildung 5.20(b) dargestellt. Bis auf die Ausreißer am Ende der Grafik, die auf numerische Ungenauigkeiten zurückzuführen sind, entsprechen die Abbildungen 5.20(a) und 5.20(b) genau den Angaben in [80].

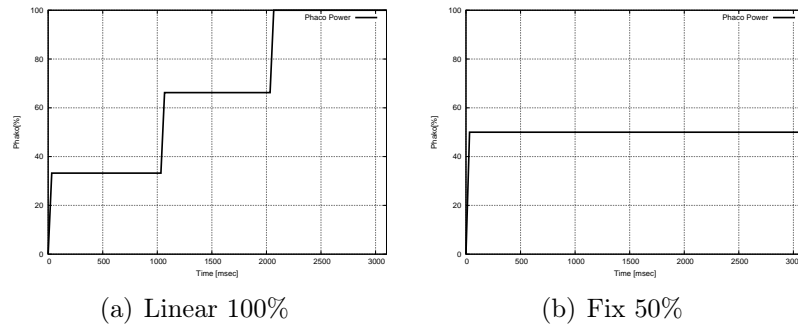


Abbildung 5.21: Der Unterschied zwischen linearer und Fixer Ultraschallmodulation.

**Vergleich Linear- und Fix-Betrieb** Für den Vergleich von Linear- und Fix-Betrieb wurde ebenfalls eine Messung über drei Sekunden durchgeführt. Der Fußpedalwert wurde dabei schrittweise um ein Drittel des Gesamtwertes erhöht. Das Maximum für die Messung mit Linear-Betrieb betrug 100% und 50% für die Messung bei Fix-Betrieb. In Abbildung 5.21(a) erhöht sich jeweils nach einer Sekunde der Ultraschallwert entsprechend der Fußpedalstellung. In Abbildung 5.21(b) bleibt der Wert immer gleich, obwohl der Fußpedalwert erhöht wird. Im Gegensatz zur Theorie kann die OMS den Ultraschallwert nicht unendlich schnell schalten. Der Schaltübergang beträgt ein Simulationsframe. Dies entspricht bei Eyesi  $\frac{1 \text{ sec}}{30 \text{ Frames}}$  ungefähr 33 ms. Aus dieser Framezeit ergibt sich die Flanke im Diagramm. Auf die Simulation hat dies keinen Einfluss, da sie nur auf diskreten Zeitpunkten arbeitet.

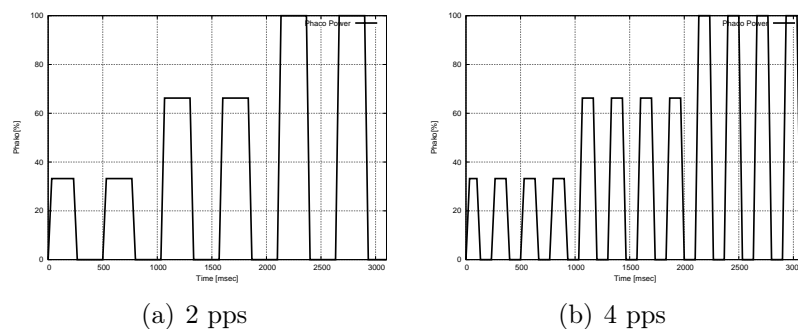


Abbildung 5.22: Die Pulse Mode Ultraschallmodulation für eine unterschiedliche Anzahl von Pulsen pro Sekunde.

**Pulse Mode** Für den Pulse Mode wurde eine Messung mit 2 und 4 Pulsen pro Sekunde (pps) durchgeführt. Die Fußpedalwerte wurden wie bei den vorherigen Messungen erhöht. Vergleicht man Abbildung 5.22(a) und 5.22(b) wird deutlich, dass die Gesamtaktiv- und inaktivzeit gleich geblieben sind. Es variiert nur die Pulsbreite. Im Vergleich mit den theoretischen Werten sind wieder Unterschiede an der Flankensteilheit zu erkennen. Diese können wie schon zuvor vernachlässigt werden.

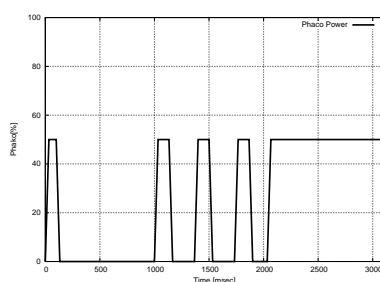


Abbildung 5.23: Die Burst Mode Ultraschallmodulation. Abhängig vom Fußpedalwert nimmt die Anzahl der Pulse zu.

**Burst Mode** Für den Burst Mode (Abb. 5.23) wurde eine Messung im Fix-Betrieb mit 50% Ultraschallleistung durchgeführt. Der Fußpedalwert wurde wieder über drei Sekunden hinweg erhöht. Wie in der Theorie nimmt die Anzahl der Pulse kontinuierlich zu, bis bei Vollausschlag der Burst Mode in den Kontinuierlichen Mode über geht.

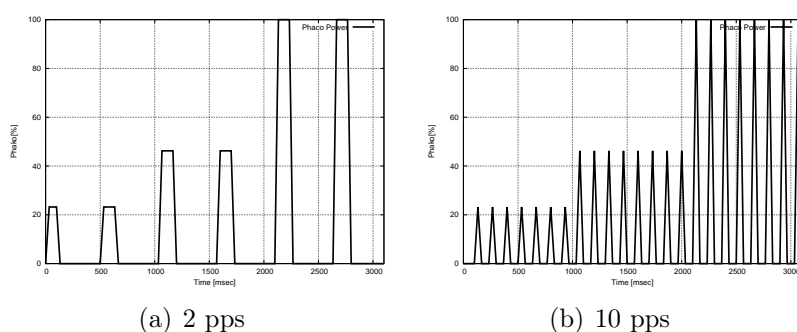


Abbildung 5.24: Die Hyperpulse Ultraschallmodulation für eine unterschiedliche Anzahl von Pulsen pro Sekunde.

**Hyperpulse** Für den Hyperpulse wurde eine Messung mit 2 und 10 Pulsen pro Sekunde durchgeführt. Der Fußpedalwert wurde wie zuvor erhöht. In Abbildung 5.24(a) und 5.24(b) ist die für den Hyperpulse geringere Gesamtaktivzeit gegenüber dem Pulse Mode zu erkennen. Bis auf die Flankensteilheit sind keine Unterschiede zur Theorie erkennbar. Erwähnenswert ist, dass für eine hohe Anzahl an Pulsen die Aktivzeit pro Pulse sehr gering wird (siehe Abbildung 5.24(b)). Sie liegt bei 10 Pulsen knapp unter dem Simulationszeitschritt. Für die Simulation stellt dies kein Problem dar, da nur mit der quantitativen Ultraschallenergie gerechnet wird.

#### 5.5.4 Vitrektomie

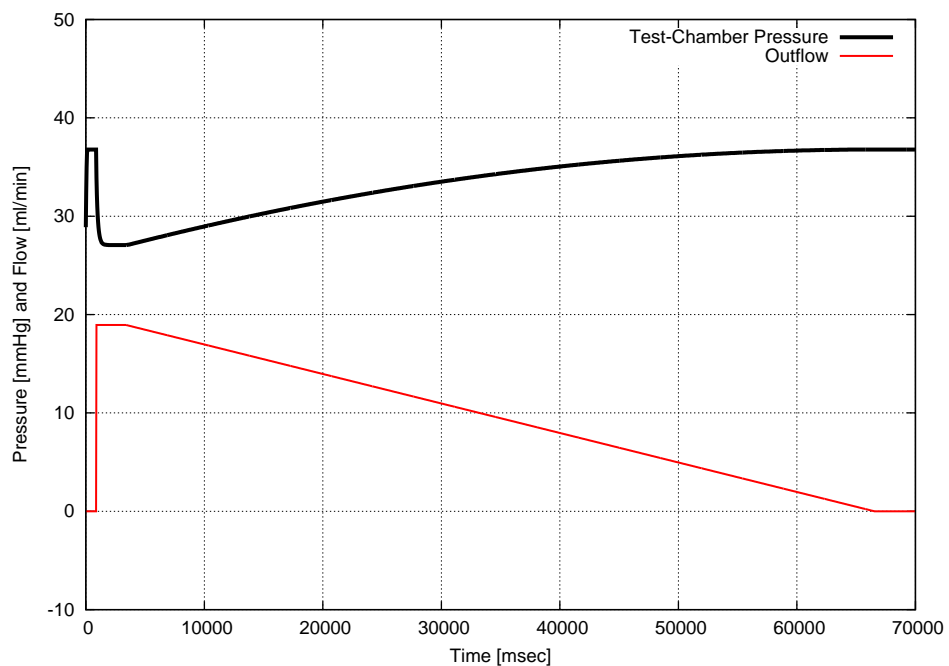


Abbildung 5.25: Die Flowrate und der Druck in der Test-Chamber in Abhängigkeit der Vitrektorschnittrate.

**Verhältnis von Schnittrate zu Fluss** In Abbildung 5.25 ist das Verhältnis vom Fluss an der Instrumentspitze und Augeninnendruck dargestellt. Die OMS war folgendermaßen konfiguriert: Flaschenhöhe 50 cm, Venturi-Pumpe im Fix-Betrieb mit 250 mmHg Vakuumlimit und linearer CFR-Wert (0.01) für die Flussreduzierung des Vitrektors. Zuerst wurde die Irrigation aktiviert. Dies ist deutlich am Anstieg des Augeninnendrucks zu erkennen. Danach

wurde die Pumpe aktiviert. Dies zeigt sich im Druckabfall und am Anstieg der Flussrate. Anschließend wurde die Schnittrate sukzessiv von 0 auf 2000 Schnitte pro Minute erhöht. Dabei ist deutlich der lineare Abfall der Flussrate zu erkennen. Der Druck in der Vorderkammer steigt an. Dies ergibt sich aus der Compliance-Funktion für den Augeninnendruck.

## 5.6 Diskussion

Bisher gab es keine Veröffentlichung zum Thema der Integration real existierender Operationsmaschinen in die virtuelle Realität. Die hier vorgestellte OMS bildet die Vorgänge auf makroskopischer Ebene nach. Dabei werden viele Vereinfachungen getroffen. Es hat sich gezeigt, dass gute Ergebnisse im Vergleich zur vorhandenen Fachliteratur und zu realen Maschinen erzielt werden.

Bei der Modellierung der OMS wurde auf Erweiterbarkeit geachtet, so dass neue reale Entwicklungen schnell in die virtuelle Realität übertragen werden können. Ein Beispiel dafür ist die Trennung in GUI-, Maschinen- und Instrumentwerte. Dadurch ist es möglich, die Berechnungen der einzelnen Werte auszutauschen, ohne direkten Einfluss auf die Simulationen in der virtuellen Augumgebung zu nehmen.

Durch die Parametrisierung ist es möglich, die Eigenschaften des Schlauchsystems und der Instrumente anzupassen.

Es sind noch nicht alle real vorkommenden Eigenschaften modelliert worden. Dies betrifft hauptsächlich die Möglichkeiten zur Surge-Unterdrückung. Dafür setzten die Maschinenhersteller spezielle mechanische Anpassungen oder Software-Lösungen für die Maschinenregelung ein. Als mechanische *Surge*-Unterdrücker können hier der **Cruise Control Filter** und **ABS-Tips** angeführt werden.

Der Cruise Control Filter wurde von STAAR<sup>®</sup> Surgical eingeführt. Bei diesem Filter handelt es sich um einen Drahtzylinder-Filter, der zwischen dem Handstück und der Aspirationsleitung angebracht wird. Der Filter verhindert, dass Linsenfragmente die Aspirationsleitung verstopfen, die Deformation der Leitung wird dadurch verringert und somit auch der Surge.

Die ABS-Tips von Alcon besitzen kleine Bohrungen im Schaft, wodurch ein ständiger Fluss ermöglicht wird. Dies führt zu einer Druckentlastung der Aspirationsleitung.

Als Software-Lösung kann das Venting, die erste maschinenseitige Möglichkeit den Surge zu verringern, angeführt werden. Beim Venting wird, sobald



der Fußpedalbereich **B** verlassen wird, Flüssigkeit oder Luft in die Aspirationsleitung abgegeben, um den Druckunterschied schneller abzubauen.

Neuere Software-Lösungen versuchen eine Verstopfung zu detektieren, um dann das maximale Vakuum automatisch zu korrigieren. Ein solches Verfahren ist CASE von AMO, das in der SOVEREIGN<sup>®</sup> COMPACT und WHITESTAR Signature<sup>™</sup> eingesetzt wird.

Für die OMS wurde ein zeitabhängiges Modell für die Ultraschallmodulation entwickelt, das sich über Aktiv- und Inaktivzeiten und Pulsraten steuern lässt. Das Modell erlaubt, dass die Pulsdauer die Framezeit der Simulation unterschreitet, da nur mit einer fiktive Ultraschallenergie pro Simulationsschritt gerechnet wird.

Das Temperaturverhalten der Phakonadel wird in diesem Modell nicht berücksichtigt. Seit Einführung der Ultraschallmodulation sind Wundverbrennungen ein eher selteneres Problem. Die Entwicklung eines entsprechenden Modells wäre in Zukunft erstrebenswert.

Für Operationen am hinteren Augenabschnitt wurde eine Modell für den Vitrektor entwickelt. Dabei wurde auf den Einfluss der Schläuche verzichtet und nur die Schnittrate berücksichtigt. Das Modell berücksichtigt den Einfluss der Schnittrate mit Hilfe eines Durchschnittswertes der Schnittrate. Um bessere Ergebnisse zu erzielen, müsste der Fluss vom aktuellen Öffnungsgrad des Instruments abhängig sein. Das Modelle liefert dennoch bereits einen plausiblen Eindruck.

Im nächsten Kapitel wird behandelt, wie die von der OMS berechneten Werte in der virtuellen Augumgebung weiter verarbeitet werden.



## Gewebeinteraktion mit dem Instrumentarium

Im vorherigen Kapitel wurde die Arbeitsweise der Maschine näher erläutert und die mathematische Umsetzung der einzelnen Eigenschaften beschrieben. In diesem Kapitel werden Modelle für die Auswirkungen der Maschinenwerte auf die virtuelle Realität entwickelt. Es werden zunächst die Abläufe in der Vorderkammer, speziell bei der Simulation der Phakoemulsifikation, genauer untersucht. Weiterhin werden Modelle für das Gewebeverhalten der virtuellen Linse entwickelt. Am Ende des Kapitels wird das Druck- und Volumenmodell auf den hinteren Augenabschnitt übertragen.

**Operationsaufbau** Wie bereits erwähnt, findet die Operation im vorderen Augenabschnitt statt. Abbildung 6.1 zeigt den Aufbau bei einer Phakoemulsifikation mit Phakomaschine, Handstück und Vorderkammer. Die Linse wird dabei mit einem aktiven Instrument aus dem Auge entfernt. Als aktiv werden Instrumenteigenschaften bezeichnet, die sich von außerhalb steuern lassen. Diese Eigenschaften sind beim Phakohandstück der Fluss, das Vakuum und der Ultraschall. Neben diesen aktiven Eigenschaften kommt es zu mechanischen Interaktionen mit den Instrumenten und der Umgebung. Bei der Modellbildung muss unbedingt auf die Einhaltung der Echtzeitbedingung geachtet werden. Dies erfordert Modelle, die trotz der beschränkten Rechenzeit ein plausibles Verhalten aufweisen. Die hier vorgestellten Lösungsansätze beziehen sich hauptsächlich auf die Simulation der Phakoemulsifikation wie sie in *Kapitel 8 - Trainingsmodule* vorgestellt wird.

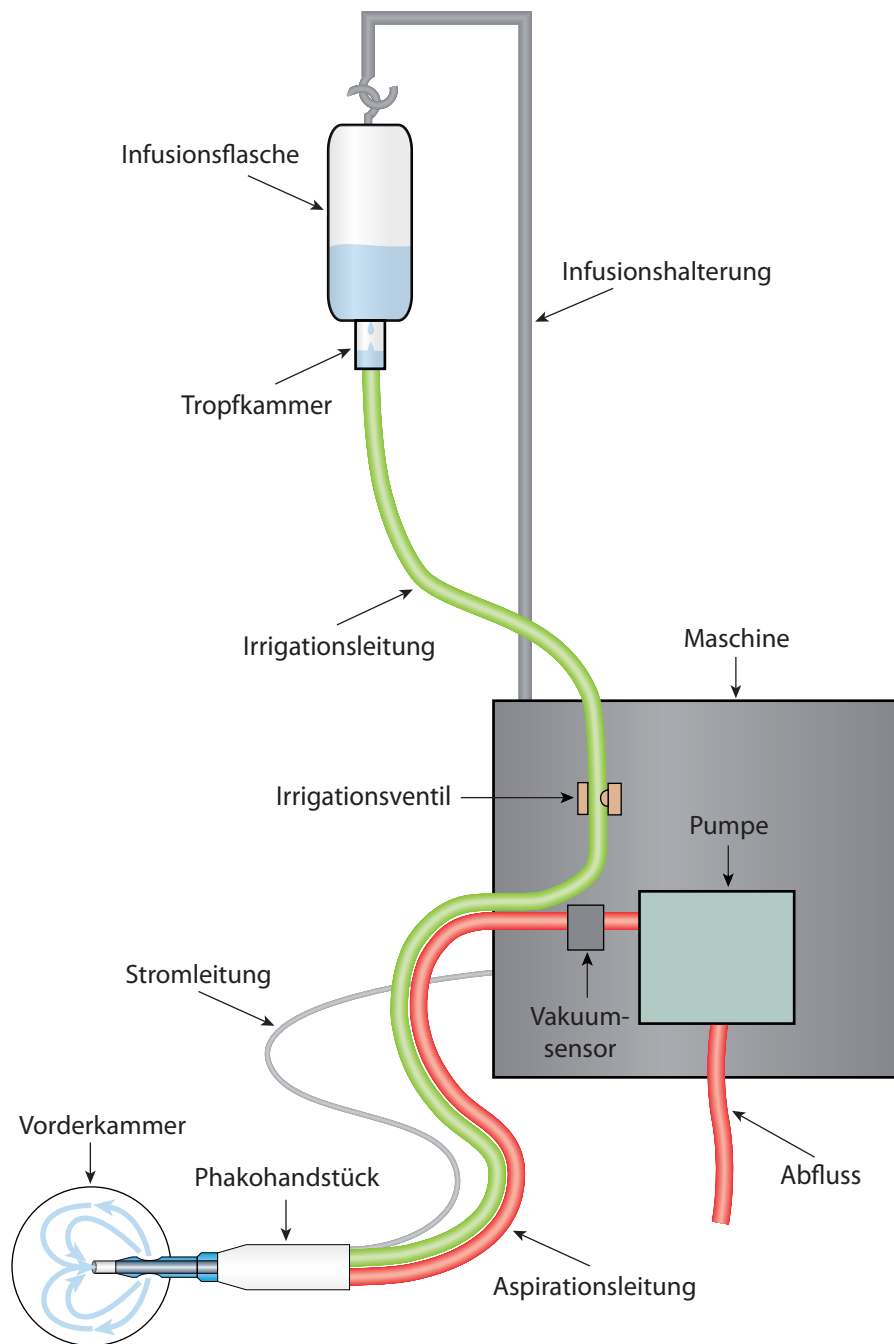


Abbildung 6.1: Der Gesamtaufbau während der Operation mit Operationsmaschine, Phakoinstrument und Vorderkammer.

## 6.1 Mechanische Interaktion

Zu Beginn werden die rein mechanischen Interaktionen untersucht. Damit ist das Verhalten der Umgebung auf die Instrumente gemeint. Die Instrumente sind virtuelle Nachbildungen der direkt ins Auge eingeführten Instrument-Dummies. Sie folgen deren Bewegungen. Eine Kollisionsbehandlung zwischen den virtuellen Instrumenten ist nicht nötig, sie überschneiden sich auch in der Realität nicht.

Es muss jedoch eine Kollisionsbehandlung mit den simulierten Objekten durchgeführt werden. Das Phakohandstück besteht in der Realität aus Metall mit einer Silikonhülle. Als zweites Instrument wird ein passives metallisches Instrument benutzt. Die Instrumente können als starre Objekte in die virtuelle Szene integriert werden. Sie dürfen per Definition nicht vom simulierten Gewebe durchdrungen werden.

Allgemeine Konzepte zur Kollisionserkennung und -behandlung wurden im *Kapitel 3 - Grundlagen der Simulation* behandelt. Wie dort erwähnt, gibt es keine allgemeingültige „optimale Lösung“ für dieses Problem.

Bei der Simulation der Phakoemulsifikation gibt es einige Besonderheiten, die bei der Kollisionserkennung berücksichtigt werden müssen.

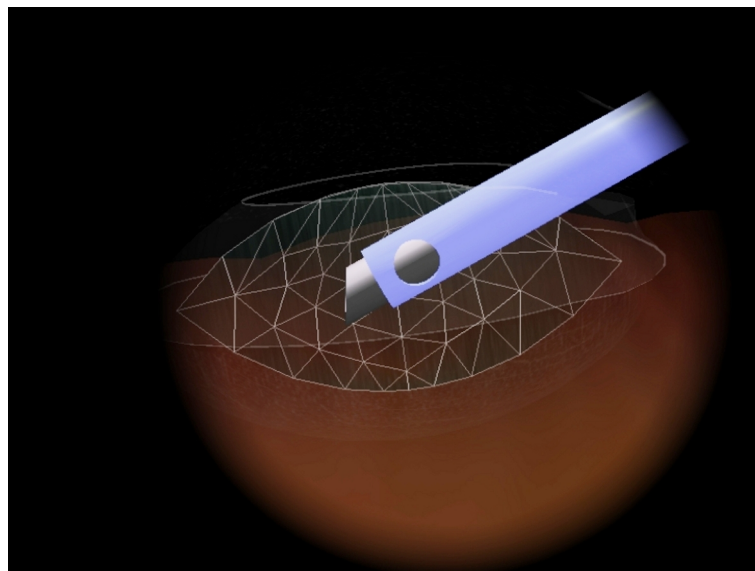


Abbildung 6.2: Das Missverhältnis von Instrumentgröße zu Mesh-Auflösung beim Querschnitt durch Auge und Linse mit Phakoinstrument.

**Missverhältnis von Instrumentgröße zu Mesh-Auflösung** Abbildung 6.2 zeigt das starke Missverhältnis von Instrumentgröße zu Mesh-Auflösung. Im Vergleich zum Instrument ist ein einzelner Tetraeder sehr groß. Dieses ergibt sich aus den Vorgaben der Echtzeiteinhaltung der Eyesi-Software. Für die Simulation wird die Linse als dreidimensionales Objekt auf Basis eines Tetraedernetzes modelliert. Dabei ist die Auflösung des Tetraedernetzes durch die Größe der realen Linse und der zur Verfügung stehenden Rechenzeit vorgegeben. Aus diesen Prämissen ergibt sich die Größe der einzelnen Tetraeder. Die Geometrie des Phakoinstruments ist durch die reale Form gegeben.

Dieses für die Simulation ungeeignete Verhältnis von Instrument- zu Tetraedergröße erfordert eine geeignete Kollisionsbehandlung.

**Kollisionserkennung und -auflösung** Für die Kollisionsbehandlung zwischen Linse und Phakoinstrument wurde die Kollisionserkennung in eine Broad- und Narrow-Phase unterteilt. Zunächst werden die möglichen Kollisionsobjekte bestimmt. Danach werden diese genauer untersucht und gleichzeitig wird eine Kollisionsantwort berechnet.

Das Phakoinstrument wird dabei durch Hüllkörper approximiert. Dies sind Zylinder für den Schaft und Kugeln für die Instrumentspitze. In der Broad-Phase werden alle Tetraeder bestimmt, die sich während dem aktuellen und dem vergangenen Simulationsschritt in den entsprechenden Hüllobjekten befanden. Die so ermittelten Tetraeder dienen als Grundlage für die Berechnungen der Narrow-Phase.

Von den in der Broad-Phase ermittelten Tetraedern werden die beteiligten Knoten, Kanten und Dreiecke extrahiert. Die Kollisionsbehandlung für die Knoten wird direkt durchgeführt. Für die Kanten, Dreiecke und Tetraeder wurde ein eigener Subsamplingansatz entwickelt. Dabei werden Kräfte für kollidierende Kanten, Dreiecke und Tetraeder nicht exakt ermittelt, sondern geschätzt. Als Grundlage für die Berechnungen dienen virtuelle Knoten, die zusätzlich zu den vorhandenen Knoten erzeugt werden. Virtuelle Knoten werden für die Kanten und die Dreiecke des Tetraeders und den Tetraeder selbst erstellt. Als Position wird das arithmetische Mittel der Knotenpositionen des entsprechenden Primitives verwendet. Wird für einen virtuellen Knoten eine Kollisionskraft ermittelt, wird diese anteilmäßig auf die ursprünglichen Knoten aufgeteilt. Die Kraft für die realen und virtuellen Knoten wird in radialer Richtung erzeugt, so dass sie über die Penalty-Konstante den kollidierenden Knoten aus dem Instrument herausbewegt.

Dies liefert zwar keine korrekten Kollisionspunkte für die Simulation, erspart jedoch die Rechenschritte, die für eine exakte Berechnung benötigt würden. Bei der Penalty-Methode handelt es sich ebenfalls um ein einfaches, physikalisch nicht korrektes Modell. Allerdings können Unzulänglichkeiten der

Kollisionserkennung und -auflösung verschmerzt werden, solange die Immersion durch eine plausible Reaktion erhalten bleibt.

Dieser Ansatz wird ebenfalls für das zweite ins Auge eingeführte Instrument und für die Kollisionen zwischen Linse und Kapsel bzw. Cornea verwendet.

## 6.2 Irrigation und Aspiration

In diesem Abschnitt werden die Aufgaben und die Auswirkungen der Irrigation und der Aspiration bei einer realen Operation untersucht und anschließend auf ein für die virtuelle Realität geeignetes Modell übertragen.

Wie in *Kapitel 5 - Physik der Phakoemulsifikation - Die Operationsmaschine* erläutert wurde, werden in der OMS die für die Operation notwendigen Werte für Irrigation und Aspiration berechnet.

Die Irrigation ist für die Erhaltung der Stabilität der Vorderkammer durch die Bereitstellung von genügend Flüssigkeit verantwortlich. Die Aufgaben der Aspiration sind die Anhaftung der Linsenstücke, so dass diese mit Ultraschall verflüssigt und abtransportiert werden können. Dies wird durch zwei unterschiedliche Effekte erzeugt, dem Aspirationsfluss (kurz Fluss) und dem Aspirationsvakuum (kurz Vakuum). Der Fluss ist für die Bewegung innerhalb der Vorderkammer verantwortlich. Durch ihn bewegen sich Objekte, die sich in der Flüssigkeit befinden, zur Instrumentspitze. Sind diese Objekte so groß, dass sie die Instrumentspitze verstopfen können, ist das Vakuum die treibende Kraft hinter der Anhaftung. Das gesamte Verhalten wird als *Followbilty* bezeichnet (siehe [80]).

Die Auswirkungen der Irrigation und Aspiration auf die Vorderkammer können in zwei Gruppen eingeteilt werden. Die eine spiegelt das gesamte Druck- und Volumenverhalten der Vorderkammer wider, die andere das Strömungsverhalten. Hier sollen diese beiden getrennt von einander untersucht werden.

### 6.2.1 Druck- und Volumenmodell der Vorderkammer

Während im *Kapitel 5 - Physik der Phakoemulsifikation - Die Operationsmaschine* das Druck- und das Flussverhalten der Maschine untersucht wurde, sollen hier die Auswirkungen der in der Maschine berechneten Werte auf die Vorderkammer behandelt werden.

Dem Autor ist kein veröffentlichtes mathematisches Druck- und Volumenmodell der Vorderkammer bekannt, das in einer Echtzeit-Simulation eingesetzt wird. Die meisten Untersuchungen sind wie bei [80] theoretischer oder experimenteller Natur.

So wird in [65] der intraokulare Augendruck (engl. *intraocular pressure*, kurz IOP) während einer Vitrektomie gemessen und mit einem Modell verglichen. Die einzige Aussage, die über das Modell getroffen wird, ist: „The indirect IOP was calculated with the IOP model according to well-known physics principles“. Dies lässt keinerlei Rückschlüsse auf das Modell zu.

In [11] wird ein mathematisches Modell für die Berechnung des Augeninnendrucks durch die Maklakov-Methode vorgestellt. Dabei wird ein Zusammenhang zwischen Elastizität der Cornea bzw. Sclera, der Volumenänderung und dem Augeninnendruck hergestellt. Das Modell ist auf die durch die Tonometrie<sup>1</sup> ermittelten Daten angepasst.

Im Folgenden wird ein vom Autor entwickeltes Druck- und Volumenmodell der Vorderkammer vorgestellt.

Zur Berechnung der Druck- und Volumenverhältnisse in der Vorderkammer wurde ein flussabhängiges Modell entwickelt. Dabei wird eine Rückkopplung über die elastische Vorderkammer berücksichtigt. Die Rückkopplung wird mit Hilfe der Compliance, dem Zusammenhang zwischen Volumen und Druck in umwandeten Strukturen, erzielt. Die Compliance wurde im letzten Kapitel (siehe Abschnitt 5.4.3.5) eingeführt.

Das Auge bzw. die Vorderkammer kann als ein mit Flüssigkeit gefülltes deformierbares Objekt mit einer elastischen Hülle angesehen werden. Genau dieser Zusammenhang wird durch die Compliance beschrieben. Für den unbelasteten Zustand besteht ein definiertes Volumen-Druck-Verhältnis. Ändert sich das Volumen, führt dies zu einer Deformation der elastischen Hülle. Die dadurch in der Hülle aufgebaute Spannung verändert den Druck im Innern des Objektes. So herrscht, bevor mit der Operation begonnen wird, für ein konstantes Augenvolumen ein Augeninnendruck von ca. 24 mmHg (3199 Pa). Dieser Innendruck ist nötig, um die Form des Auges aufrecht zu erhalten. Während der Operation variiert der Innendruck, bedingt durch Zu- und Abfluss von Flüssigkeiten. Dieses Verhalten wird mit dem folgenden Modell nachgebildet.

Dafür werden die Auswirkungen der Irrigation und der Aspiration auf die Druck- und Volumenverhältnisse der Vorderkammer untersucht. Bei den folgenden Betrachtungen wird davon ausgegangen, dass es sich um ideale Flüssigkeiten handelt. Es werden die durch die Maschinen-Simulation berechneten Irrigations- und Aspirationswerte verwendet.

**Zufluss** Die Irrigation ist der einzige Zufluss von Flüssigkeit in die Vorderkammer bzw. ins Auge. Dabei fließt die Flüssigkeit durch die beiden Irrigationsöffnungen am Irrigationsaufsatz. Die Flaschenhöhe bestimmt den

---

<sup>1</sup>Das übliche Verfahren zur Messung des Augeninnendrucks.



Druck der Wassersäule. Wie im vorherigen Kapitel behandelt, bewirkt dieser Druck eine Strömung an der Instrumentspitze. Die Irrigationsöffnungen sind wesentlich kleiner als der Schlauchdurchmesser, deshalb können die Reibungsverluste durch die Irrigationsschlauchlänge vernachlässigt werden. Im Allgemeinen gilt für die Strömungsgeschwindigkeit  $v$  beim Ausströmen aus einem Loch:

$$v = \sqrt{\frac{2p}{\rho}} \quad (6.1)$$

mit der Dichte  $\rho$  der Flüssigkeit und dem Druck  $p$ . Bei reinem Schweredruck gilt mit der Erdbeschleunigung  $g$  und der Höhe  $h$  der Wassersäule:

$$v = \sqrt{2gh} \quad (6.2)$$

Damit lässt sich der Volumenstrom  $\dot{V}$  für die Querschnittsfläche  $A$  berechnen

$$\dot{V} = A * v = A * \sqrt{\frac{2p}{\rho}} \quad (6.3)$$

Bei einer freien Strömung ist nur die Flaschenhöhe ausschlaggebend für die Strömungsgeschwindigkeit. Bei einem gekoppelten System, wie es hier betrachtet wird, spielt der Gegendruck auf die Irrigationsöffnung eine wichtige Rolle. Die wirkende Druckdifferenz ergibt sich aus dem Augeninnendruck und der Flaschenhöhe. Während der Operation wirkt nicht der reine Schweredruck, sondern die Druckdifferenz  $\Delta p$  an den Öffnungen des Irrigationsaufsatzes:

$$\Delta p = p_{Instrument} - p_{IO} \quad (6.4)$$

wobei  $p_{Instrument}$  der Schweredruck an der Instrumentspitze, der durch die Flaschenhöhe erzeugt wird, und  $p_{IO}$  der aktuelle Augeninnendruck ist.

Dadurch ergibt sich für das einströmende Volumen:

$$\dot{V}_{Instrument} = A * \sqrt{\frac{2(p_{Instrument} - p_{IO})}{\rho}} \quad (6.5)$$

Nachdem sich ein Gleichgewicht an der Irrigationsöffnung eingestellt hat, kommt der Zufluss zum Erliegen und es herrscht der Schweredruck, der über die Flaschenhöhe erzeugt wird. Über die Irrigation wird direkt der Druck und indirekt das Volumen in der Vorderkammer beeinflusst.

**Abfluss** Der Abfluss besteht aus mehreren Komponenten: der Aspiration, der Leckage (engl. *Leakage*) an den Instrumentzugängen und dem natürlichen Fluss im Auge. Dabei wird die Aspiration aktiv von der Maschine (siehe *Kapitel 5 - Physik der Phakoemulsifikation - Die Operationsmaschine*) erzeugt. Sie entfernt Flüssigkeit und Linsenmaterial aus der Vorderkammer und wirkt sich direkt auf das Volumen der Vorderkammer und somit indirekt auf den Druck aus. Die Leckage berücksichtigt den Abfluss an den Zugängen durch Verkippungen der Instrumente. Der Volumenstrom für die Leckage lässt sich über den Druckunterschied und die Öffnungsgröße des Zugangs berechnen. Als ausgleichender Anteil wurde bei erhöhtem Druck ein sehr geringer unabhängiger natürlicher Fluss im Auge berücksichtigt.

Für den gesamten Abfluss gilt:

$$\dot{V}_A = \dot{V}_{\text{Asp}} + \dot{V}_{\text{Leck}} + \dot{V}_{\text{Nat}} \quad (6.6)$$

mit dem Aspirationsvolumenstrom  $\dot{V}_{\text{Asp}}$ , der Leckage  $\dot{V}_{\text{Leck}}$  und dem natürlichen Fluss  $\dot{V}_{\text{Nat}}$ .

**Flussdifferenz** Aus dem Zu- und Abfluss lässt sich der gesamte Volumenstrom  $\Delta\dot{V}$  durch die Vorderkammer berechnen:

$$\Delta\dot{V} = \dot{V}_Z - \dot{V}_A \quad (6.7)$$

Mittels zeitlicher Integration wird das neue Volumen berechnet. Dabei wird der Volumenstrom während des Zeitschrittes als konstant angenommen. Somit bietet sich eine explizite Berechnung an:

$$V_{t+1} = V_t + \Delta\dot{V} * \Delta t \quad (6.8)$$

**Druckberechnung** Über die Compliance wird das elastische Verhalten der Vorderkammer modelliert:

$$p_{IO} = C(V) \quad (6.9)$$

Dabei ist  $C(V)$  eine vom Volumen abhängige skalare Funktion, die für das Normalvolumen den Normaldruck zurückliefert, für größere Volumen einen größeren Druck und für geringere Volumen einen geringeren Druck. Die Compliance-Funktion wurde so gewählt, dass sich Abweichungen vom Normalvolumen  $V_n$  zum Normaldruck  $p_n$  getrennt voneinander einstellen lassen. Es

wurden zwei Compliance-Funktionen entwickelt und untersucht: eine lineare (Gleichung 6.10) und eine nichtlineare (Gleichung 6.11).

Die lineare Compliance-Funktion wurde als einfache Approximation gewählt. Sie lässt sich über die vier Parameter für die Volumenobergrenze  $V_o$ , die Volumenuntergrenze  $V_u$ , die Druckobergrenze  $p_o$  und die Druckuntergrenze  $p_u$  einstellen.

$$C_{\text{lin}}(V) = \begin{cases} \frac{V_u - V_n}{p_u - p_n} * (V - V_n) + p_n, & \text{wenn } V < V_n \\ \frac{V_o - V_n}{p_o - p_n} * (V - V_n) + p_n, & \text{sonst} \end{cases} \quad (6.10)$$

Für die nichtlineare Compliance-Funktion wurde der  $\text{artanh}$ , die Umkehrfunktion des  $\tanh$ , gewählt, da der  $\tanh$  eine gute Näherung zur in der Literatur vorkommenden Compliance-Funktion (wie in Abbildung 5.10) darstellt. Zusätzlich lässt sich die Steilheit über nur einen weiteren Parameter anpassen.

$$C_{\text{non-lin}}(V) = \begin{cases} \text{artanh}\left(\frac{V - V_u}{V_n - V_u} - 1\right) * m_u + p_n, & \text{wenn } V < V_n \\ \text{artanh}\left(\frac{V - V_n}{V_o - V_n}\right) * m_o + p_n, & \text{sonst} \end{cases} \quad (6.11)$$

mit den zusätzlichen Steigungen  $m_u$  und  $m_o$  für die Steilheit der Funktion. Im Gegensatz zur linearen Version ist die nichtlineare nicht durch Druckober- oder -untergrenzen beschränkt, sondern steigt asymptotisch an den Volumengrenzen an.

**Der dynamische Druck** Abschließend wird der statische Druck um den dynamischen Druck der strömenden Flüssigkeit reduziert. Danach liegt der neue Druckwert  $p_{IO}$  der Vorderkammer vor.

**Ablauf eines Simulationsschrittes** Zusammengefasst ergeben alle Berechnungen für das Voderkammerverhalten den Algorithmus 3.

In dem hier vorgestellten Algorithmus 3 werden alle Werte, so wie sie zuvor behandelt wurden, der Reihe nach abgearbeitet. Über die Maschinen-Simulation werden implizit die Auswirkungen der Okklusion und eines erhöhten Aspirationsflusses durch den Surge berücksichtigt.

**Algorithmus 3** Druck- und Volumenberechnungen für die Vorderkammer

---

```
1: for all  $i$  solange  $i \leq n$  do
2:    $\Delta p = p_{Instrument} - p_{IO}$ 
3:   if  $\Delta p > 0$  then
4:     berechne  $\dot{V}_{Instrument}$ 
5:   end if
6:   if  $p > 0$  und aktivierte Irrigation then
7:     berechne  $\dot{V}_{Leck}$ 
8:   end if
9:   if  $p >$  dem Druck der natürlichen Flussgrenze then
10:    berechne  $\dot{V}_{Nat}$ 
11:  end if
12:  Abfragen des Aspirationsfluss  $\dot{V}_{Asp}$  an der Instrumentspitze
13:  berechne  $\Delta \dot{V} = \dot{V}_{Instrument} - (\dot{V}_{Asp} + \dot{V}_{Leck} + \dot{V}_{Nat})$ 
14:  berechne  $V_{t+1} = V_t + \Delta \dot{V} * \frac{\Delta t}{n}$ 
15:  berechne  $p = C(V)$ 
16:  reduziere  $p$  und den dynamischen Druck der strömenden Flüssigkeit
17: end for
```

---

### 6.2.2 Das Flussfeld - Auswirkungen auf die Umgebung

Die Irrigation und Aspiration, die durch die Maschine erzeugt werden, wirken sich auf das Strömungsverhalten in der Vorderkammer aus. Die Vorderkammer ist mit Flüssigkeit gefüllt, die bei aktivierter Aspiration zu strömen beginnt. Dabei fließt die Flüssigkeit immer von einer Quelle zu einer Senke. In Abbildung 6.3 ist das Flussfeld, das durch ein Standard-Phakohandstück erzeugt wird, dargestellt, wie es in der Theorie verwendet wird [80] und [4].

Die Irrigationslösung fließt von den beiden Irrigationsöffnungen (Quellen) zur Instrumentspitze (Senke). Die Zugänge zum Auge bilden weitere Senken, aus denen Flüssigkeit entweichen kann. Über sie entweicht jedoch nur ein sehr geringer Teil.

Die Strömung in der Vorderkammer wirkt sich über die konstante *Mobilität*  $\mu$  auf die darin enthaltenen Objekte aus und transportiert diese mit der Flüssigkeit weiter.

Das Flussfeld, das in der Vorderkammer herrscht, kann über verschiedene Modelle erzeugt werden. Es kann deskriptiv oder physikalisch basiert modelliert werden.

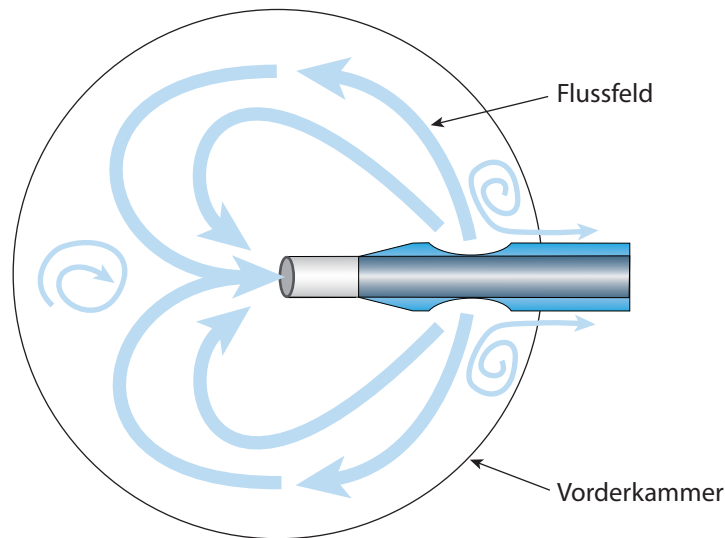


Abbildung 6.3: Das Flussfeld, das durch das Phakoinstrument in der Vorderkammer erzeugt wird.

#### 6.2.2.1 Deskriptive Modellierung des Flussfeldes

Bei der deskriptiven Modellierung des Flussfeldes wird um das Instrument ein Kraftfeld erzeugt. Es wurde ein einfaches Modell entwickelt, das versucht, ein Flussfeld ähnlich dem aus Abbildung 6.3 zu erzeugen. Die Analyse von Operationsvideos diente als weitere Quelle für die Modellierung.

In Abbildung 6.4 sind die Koordinatenachsen des Phakoinstruments und des Weltkoordinatensystems dargestellt. Das Phakoinstrument ist entlang der z-Achse (blau) ausgerichtet, die x-Achse (rot) liegt in Richtung der Irrigationsöffnung und die y-Achse (grün) orthogonal zu den beiden anderen.

Das erzeugte Kraftfeld ist in vier Bereiche aufgeteilt (siehe Abbildung 6.5), die beiden Bereiche um die Irrigationsöffnungen, einen fernen und einen nahen Aspirationsbereich.

Für die folgenden Erläuterungen wird jeweils der zu dem Kraftfeld gehörende Bereich des Phakoinstruments gewählt, so dass die verwendeten Vektoren eindeutig bezeichnet werden können.

Für den Bereich um die Irrigationsöffnungen wird für jede Hälfte des Phakoinstruments ein halbkugelförmiges Feld erzeugt, dessen Ausprägung über seinen Radius bestimmt wird (siehe Abbildung 6.6). Die auf das Objekt wirkende Kraft ist abhängig vom Aspirationsfluss und von der Entfernung zur Öffnung.

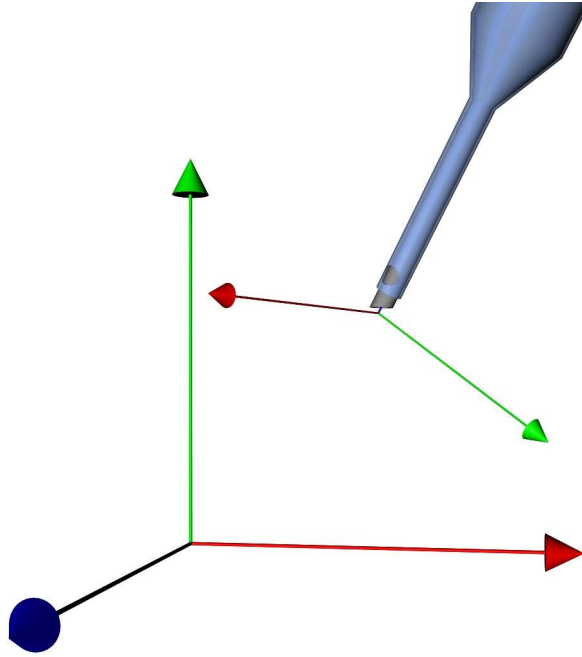


Abbildung 6.4: Die Ausrichtung der Weltkoordinatenachsen und der Koordinatenachsen des Phakoinstruments (x-Achse (rot), y-Achse (grün) und z-Achse (blau)).

Zunächst wird die Lage  $\vec{r}$  des Knotens mit der Position  $\vec{x}$  relativ zur ersten Irrigationsöffnung  $\vec{IO}^1$  bestimmt:

$$\vec{r} = \vec{x} - \vec{IO}^1 \quad (6.12)$$

Dabei werden alle Berechnungen im Koordinatensystem der Linse durchgeführt. Dies erfordert die Transformation des Instrumentkoordinatensystems in das Linsensystem.

Über die Lage lässt sich feststellen, ob der betrachtete Punkt  $\vec{r}$  innerhalb des Irrigationsradius  $r_I$  und auf der Seite der Irrigationsöffnung liegt. Dafür muss gelten:

$$|\vec{r}| < r_I \quad (6.13)$$

und

$$\vec{r} \cdot \vec{IO}_x^1 > 0. \quad (6.14)$$

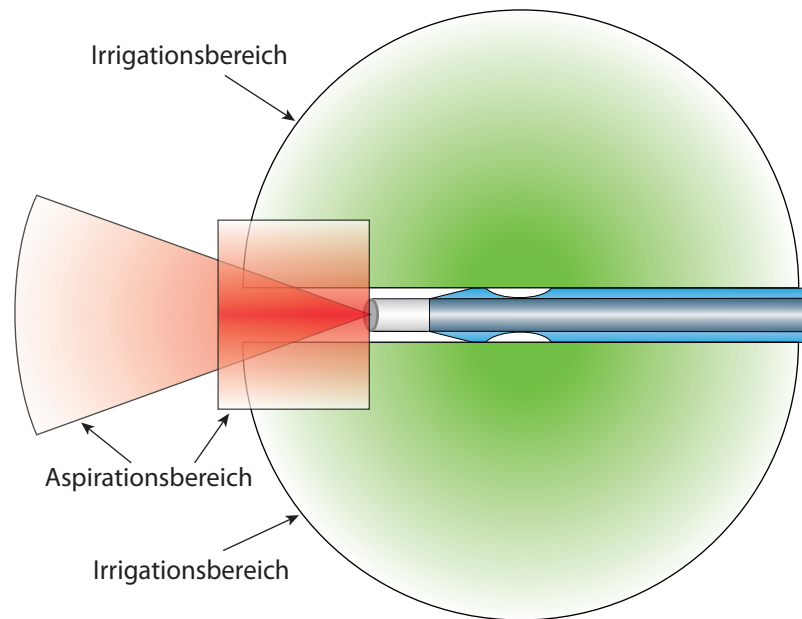


Abbildung 6.5: Die Interaktionsbereiche des Phakoinstruments. Grün dargestellt der Irrigationsbereich und rot der Aspirationsbereich.

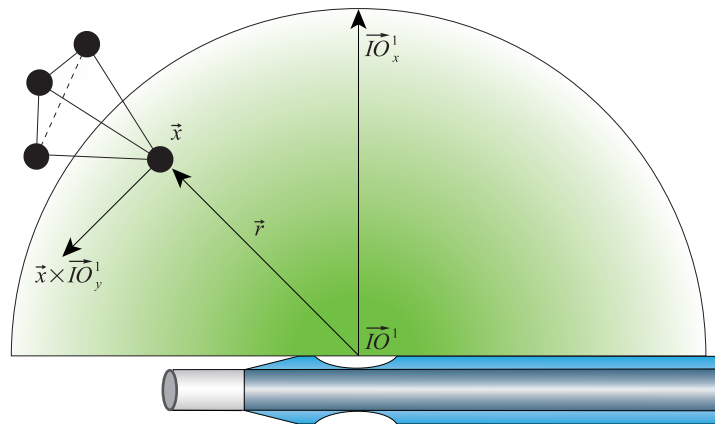


Abbildung 6.6: Ein Irrigationsbereich des Phakoinstruments.

Sind die Bedingungen in Gleichung 6.13 und 6.14 erfüllt, wird die Kraft:

$$\vec{F}_{IO^1}(\vec{r}, \text{Flow}) = \frac{\left\| \vec{r} \times \vec{IO}_y^1 \right\| * \text{Flow} * \mu_I}{|\vec{r}|^2} \quad (6.15)$$

die auf den Knoten mit der Position  $\vec{x}$  wirkt, berechnet.

Das Kreuzprodukt zwischen dem Positionsvektor  $\vec{r}$  und der transformierten Koordinatenachse  $\vec{IO}_y^1$  liefert einen zu beiden Vektoren senkrechtstehenden Vektor zurück. Dieser wird normiert, die entfernungsabhängige Skalierung fließt über den quadratischen Faktor  $|\vec{r}|^2$  in das Modell ein. Die Kraft wird über den maschinenabhängigen Flow-Wert und die vorgegebene Mobilität  $\mu_I$  kontrolliert.

Für die gegenüberliegende Irrigationsöffnung werden die Berechnungen mit angepassten Vorzeichen durchgeführt. Das so erzeugte Kraftfeld um die Irrigationsöffnungen treibt die darin befindlichen Knoten vor die Instrumentspitze. Dort werden sie über die Aspirationskräfte zur Instrumentspitze gezogen.

Für den fernen Aspirationsbereich (siehe Abbildung 6.7) wurde ein kegelförmiger Interaktionsbereich gewählt und für den nahen Aspirationsbereich (siehe Abbildung 6.8) ein zylinderförmiger.

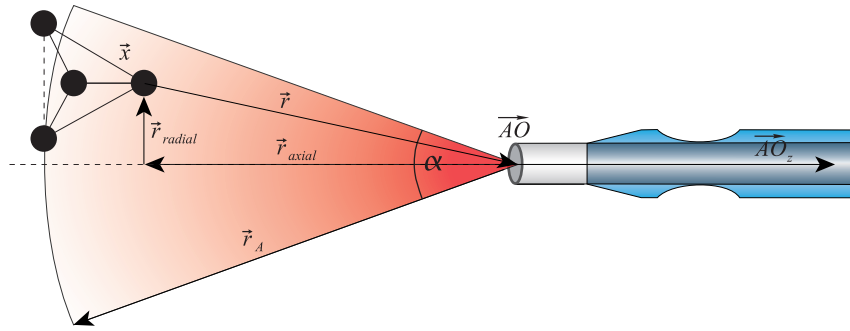


Abbildung 6.7: Der ferne Aspirationsbereich des Phakoinstruments.

Die Ausprägung des fernen Aspirationsbereichs ist festgelegt durch den Radius um die Instrumentspitze  $\vec{AO}_g$ , den Öffnungswinkel  $\alpha_{AO}$  des Saugkegels und den Halbraum vor der Aspirationsöffnung, der durch  $\vec{AO}_z$  gegeben ist:

$$\vec{r} = \vec{AO}_g - \vec{x} \quad (6.16)$$

Für die relative Lage  $\vec{r}$  zur Instrumentspitze müssen somit folgende Bedingungen erfüllt sein:

$$|\vec{r}| < r_{A_g} \quad (6.17)$$



$$\arccos \frac{\vec{r} \cdot \vec{AO}_z}{|\vec{r}| |\vec{AO}_z|} < \alpha_{AO} \quad (6.18)$$

$$\vec{r} \cdot \vec{AO}_z > 0 \quad (6.19)$$

damit der Knoten innerhalb des Aspirationsbereichs liegt.

Die Aspirationskraft  $\vec{F}_{A_g}$  auf den Knoten ergibt sich folgendermaßen:

$$\vec{F}_{A_g}(\vec{r}, \text{Flow}) = \frac{\|\vec{r}\| * \text{Flow} * \mu_{A_g}}{|\vec{r}|} \quad (6.20)$$

Dabei bezeichnet  $\|\vec{r}\|$  den normierten Vektor  $\vec{r}$  und  $\mu_{A_g}$  die Mobilität für die Aspiration. Die Aspirationskraft wird linear mit dem Abstand zur Aspirationspitze skaliert, dies erzeugt auch für weit entfernte Objekte eine ausreichend starke Bewegung zur Instrumentspitze.

Nahe an der Instrumentspitze liegende Knoten fallen nicht in den am Instrument noch relativ schmalen Öffnungsbereich des Aspirationskegels. Der ferne Aspirationsbereich wird deshalb um den nahen Aspirationsbereich ergänzt. Dafür wurde eine zylindrische Form gewählt und alle Knoten bestimmt, die darin liegen. Die nahe Aspirationskraft zeigt ebenfalls in Richtung der Aspirationsöffnung:

$$\vec{r} = \vec{AO}_n - \vec{x} \quad (6.21)$$

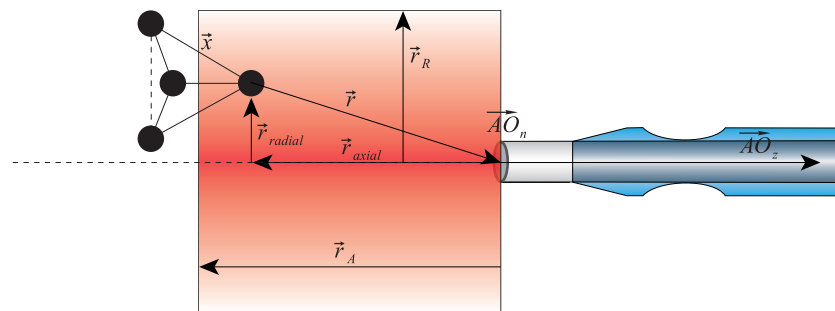


Abbildung 6.8: Der nahe Aspirationsbereich des Phakoinstruments.

Es müssen drei Bedingungen erfüllt sein, damit der Knoten mit der Position

$\vec{x}$  innerhalb des Aspirationsbereichs liegt.

Gleichung 6.19 muss erfüllt sein, damit er vor der Instrumentöffnung liegt. Für den Betrag der axialen Komponente  $\vec{r}_{\text{axial}}$  muss gelten:

$$|\vec{r}_{\text{axial}}| < r_A \quad (6.22)$$

wobei  $r_A$  die axiale Ausdehnung des Interaktionszylinders ist. Als dritte Bedingung muss für die radiale Komponente gelten:

$$|\vec{r}_{\text{radial}}|^2 < r_R \quad (6.23)$$

mit der radialen Ausdehnung  $r_R$ .

Die Kraft auf den Interaktionsknoten berechnet sich nach Gleichung 6.20. So wird für jeden Knoten die auf ihn wirkende Kraft relativ zum Instrument berechnet. Dabei werden die Kräfte für in mehreren Bereichen liegenden Knoten akkumuliert. Aus Stabilitätsgründen werden die Kräfte auf alle Knoten und nicht nur auf die Oberflächenknoten angewendet. Über die zeitliche Integration wirkt sich die Kraft auf die Bewegung der Objekte aus.

Die Kraftberechnung ist bei dieser Methode nur von der Anzahl der Knoten und der verwendeten Interaktionsbereiche abhängig. Dies ermöglicht eine hinreichend schnelle Berechnung. Ebenso ist die Kraftübertragung und -ausbreitung einfach zu verstehen und lässt sich über die Mobilität  $\mu$  einstellen.

Die Methode ist physikalisch nicht korrekt. Das Flussfeld verhält sich immer gleich und wird nicht durch die Umgebung (wie Cornea, Kapsel, Iris) beeinflusst. Die Strömung ebbt schnell ab, wenn die Aspiration und Irrigation abgeschaltet werden. Sie liefert jedoch ein visuell plausibles Verhalten der Flüssigkeitsinteraktion in der Vorderkammer. Wie alle deskriptiven Modelle erfordert es einigen Aufwand, bis alle Parameter ausreichend gut eingestellt sind.

#### 6.2.2.2 Physikalisch basierte Modellierung des Flussfeldes

Eine andere Möglichkeit ist, eine Flüssigkeitssimulation im kompletten Bereich der Vorderkammer durchzuführen. Durch Angabe der Quellen und Senken lässt sich damit das komplette Strömungsverhalten der Flüssigkeit simulieren. Wie in *Kapitel 3 - Grundlagen der Simulation* beschrieben, können partikel- oder gitterbasierte Ansätze verwendet werden. Bei partikelbasierten Ansätzen repräsentieren die Partikel selbst die Flüssigkeit. Eine Kraftübertragung von den Partikeln auf die Objekte in der Flüssigkeit erfordert eine

aufwendige Kollisionsbehandlung. Deshalb fiel die Wahl auf einen gitterbasierten Ansatz. Für jeden Gitterpunkt erhält man dabei einen Geschwindigkeitsvektor, der über die Mobilität  $\mu$  die Bewegung an die Objekte überträgt, die sich in diesem Gitterbereich befinden. Die Ermittlung des entsprechenden Gitterbereichs erfolgt nach der Methode der Spatial Subdivision. Durch Randbedingungen wird der von der Flüssigkeit eingenommene Bereich begrenzt. Die Implementierung ist eine 3D-Erweiterung des Ansatzes von Stam [88]. Die für die Simulation notwendigen Quellen und Senken sind durch die Position der Instrumentöffnungen in der Vorderkammer gegeben. Die Stärke der Strömung ergibt sich aus dem Wert für den Aspirationsfluss.

Die Vorteile der Simulation sind, dass sie physikalisch basiert ist und sich abhängig von der Instrumentposition ein realistisches Flussfeld ausbildet. Die Nachteile sind eine komplexere Implementierung, die mehr Rechenzeit und Speicherbedarf benötigt. Der große Rechenzeitbedarf reduziert die für weitere Simulationen verbleibende Rechenzeit. Dies ließ keine Echtzeitsimulation in Kombination mit der Linsensimulation zu.

Verwendet wird die Simulation des Flussfeldes im Phako-Training, einem später vorgestellten Trainingsmodul. Die im Phako-Training durchgeführten Berechnungen lassen genügend Spielraum, um eine Flüssigkeitssimulation auf der aktuellen Eyesi-Hardware zu verwenden.

## 6.3 Verstopfen der Instrumentspitze

In der Realität ergibt sich die Verstopfung der Instrumentspitze durch die Saugkräfte und die mechanische Gegenkraft des Schaftes auf das Gewebe. Das Größenmissverhältnis, das in der Simulation zwischen Simulationsgitter und Instrument besteht, erschwert die Verstopfungsberechnung. Für die Maschinen-Simulation wird der Verstopfungsgrad der Instrumentspitze benötigt. Dieser wird wie folgt berechnet:

Die Instrumentöffnung wird durch  $n$  Punkte approximiert. Diese dienen als Referenz für die Verstopfung. Befindet sich ein Punkt innerhalb eines Tetraeders wird ein  $n$ -tel der Instrumentspitze als verstopft angenommen und der beteiligte Tetraeder als verstopfend markiert. Als verstopfend markierte Tetraeder werden dann über sehr steife virtuelle Federn an die Instrumentspitze gebunden. Diese Verbindung wird gelöst, wenn das Aspirationsvakuum zu gering ist, der Tetraeder abgesaugt wird oder eine Maximallänge der Feder überschritten wird. Die so gekoppelten Knoten lassen sich mit dem Instrument mitbewegen.

## 6.4 Ablösen des Gewebes

Um die reale Linse aus der Vorderkammer zu entfernen, wird diese verflüssigt und abgesaugt. Dies muss in der virtuellen Realität nachgebildet werden. Dafür muss das simulierte Gewebe entsprechenden topologischen Änderungen unterzogen werden. Hierfür wurde die Möglichkeit gewählt, komplette Tetraeder aus dem Simulationsmesh zu lösen. Herausgelöste Tetraeder werden solange weiter simuliert bis sie abgesaugt sind. Dann werden sie aus der Simulation entfernt.

Für das Herauslösen wurden zwei Modelle entwickelt. Das eine nutzt den vom Instrument erzeugten Ultraschall, das andere das Aspirationsvakuum.

### 6.4.1 Ultraschall

Der Ultraschall wird bei Standard-Phakohandstücken durch eine longitudinale Auslenkung der Phakonadel erzeugt. Zurzeit ist den Medizinnern noch unklar, welcher Effekt hauptverantwortlich für das Verflüssigen ist. Deshalb wurde ein deskriptives Modell entwickelt, das den Ultraschalleffekt auf makroskopischer Ebene plausibel modelliert.

Es basiert auf der Idee, dass das Gewebe nur eine gewisse Menge an Ultraschall verkraften kann, bis es zerstört wird. Die reale Ultraschallenergie ist von vielen Faktoren abhängig, wie zum Beispiel von der Instrumentform, dem Schalldruck, der Schallschnelle, der Dichte der umgebenden Flüssigkeit und der Schallgeschwindigkeit der Flüssigkeit. Die meisten dieser Faktoren sind während der Simulation nicht bekannt. Für das hier vorgestellte Modell wird eine fiktive Ultraschallenergie  $E_t^{US}$  verwendet. Diese wird in der Maschine für den aktuellen Simulationszeitschritt berechnet. Sie setzt sich aus der Aktivzeit und der Auslenkung der Phakonadel zusammen (siehe *Kapitel 5 - Physik der Phakoemulsifikation - Die Operationsmaschine*). Dieser Wert wird während der Simulation verwendet, um das Gewebe zu schwächen.

Die Menge an Ultraschallenergie  $E_t^G$ , die vom simulierten Gewebe (Tetraeder) pro Zeitschritt aufgenommen wird, wird nach Gleichung (6.24) berechnet. In dieses Modell fließt die Entfernung  $d$  zur Instrumentspitze, die von der Maschine erzeugte Energie und die vom Gewebe bereits aufgenommene Energie ein.

$$E_t^G = \frac{E_t^{US}}{d^2} + E_{t-1}^G * e^{-\Delta t * \lambda}, \quad \lambda > 0 \quad (6.24)$$

Dabei bleibt die vom Gewebe aufgenommene Energie nicht dauerhaft erhalten, sondern reduziert sich wieder. Die Regenerationsdauer kann über den

$\lambda$ -Parameter eingestellt werden. Die Regeneration wurde erst später eingefügt. Bevor es diese Möglichkeit gab, hatte es den Anschein, als lösen sich Tetraeder plötzlich, sobald Ultraschall erneut auf sie wirkte.

Das Linsenmodell, das in dem später vorgestellten Trainingsmodul (siehe Kapitel 8 - *Trainingsmodule*) verwendet wird, besteht aus einem Tetraedermesh. Dabei kann für jeden Tetraeder ein maximaler Ultraschallwert  $E_{max}^G$  angegeben werden. Überschreitet die vom Gewebe aufgenommene Menge an Ultraschallenergie den Schwellwert, der für den aktuellen Tetraeder gilt, wird dieser aus dem Simulationsmesh gelöst. Er kann sich dann frei bewegen und wird bei aktivierter Aspiration in das Instrumentinnere gesaugt und aus der Simulation entfernt.

Mit diesem Modell wird auf makroskopischer Ebene versucht, das Ultraschallverhalten in die virtuelle Realität zu übertragen.

### 6.4.2 Vakuum

Befindet sich die Katarakt noch im Anfangsstadium, ist die Linse noch sehr weich und es reicht oft das Aspirationsvakuum aus, um das Linsengewebe abzusaugen. Das entwickelte Modell berücksichtigt den auf den Tetraeder wirkenden Vakuumwert ähnlich dem Ultraschallwert.

Zunächst werden alle Tetraeder bestimmt, die sich in einem maximalen Abstand um die Instrumentöffnung befinden. Diese werden mit dem aktuellen Vakuum belastet. Dieses wird auf das bereits aufgenommene Vakuum addiert. Der Tetraeder kann sich auch hier wieder regenerieren. Dafür wird das bereits aufgenommene Vakuum wieder reduziert. Für das vom Gewebe aufgenommenen Vakuum wird folgendes Modell benutzt:

$$V_t^G = V_t^{\text{Instrument}} + V_{t-1}^G * e^{-\Delta t * \lambda}, \quad \lambda > 0 \quad (6.25)$$

mit dem aktuellen Vakuumwert  $V_t^G$ , dem vom Instrument abgegeben Vakuumwert  $V_t^{\text{Instrument}}$ , dem bis dahin aufgenommenen Vakuum  $V_{t-1}^G$  und dem Regenerationsterm  $e^{-\Delta t * \lambda}$  mit der Zeitkonstante  $\lambda$ .

Zu Beginn der Simulation wird der Grenzwert für jeden Tetraeder auf einen Maximalwert  $V_{max}^G$  eingestellt. Wirkt während der Simulation Ultraschall auf einen Tetraeder wird der Grenzwert reduziert. Das Gewebe wird zusätzlich durch Ultraschall geschwächt. Ein Tetraeder wird aus dem Gewebe gelöst wenn gilt:

$$V_t^G > \frac{V_{max}^G}{1 + E_t^G}. \quad (6.26)$$

## 6.5 Absaugen des Gewebes

Während der Operation wird das Linsenmaterial durch das Instrument abgesaugt und wandert durch den Aspirationsschlauch in einen Auffangbehälter. Dies sind Bereiche, die für den visuellen Eindruck der Simulation nicht von Bedeutung sind. Die herausgelösten einzelnen Tetraeder werden deshalb nur bis zu einer vorgegeben Stelle in das Instrument gesaugt und dann endgültig aus der Simulation entfernt. Bis zum Erreichen dieser Stelle wirkt neben der Saugkraft eine Penalty-Kraft auf die einzelnen Knoten des Tetraeders, damit dieser das Instrument nicht seitlich verlassen kann.

## 6.6 Erweiterungen für den hinteren Augenabschnitt

Die Erkenntnisse über das Druck- und Volumenverhalten der Vorderkammer wurden auf den hinteren Augenabschnitt übertragen. Grundsätzlich gelten bei Operationen im hinteren Augenabschnitt die gleichen Voraussetzungen wie in der Vorderkammer. Über die Irrigation muss das Auge stabil gehalten werden und über die Aspiration wird Flüssigkeit und Gewebe aus dem Auge entfernt.

Für die Druckberechnung des hinteren Augenabschnitts wird der Algorithmus, der für die Druckberechnung der Vorderkammer entwickelt wurde, verwendet. Dieser wurde um den Spezialfall der Eindellung (engl. Indentation) erweitert. Bei einer Indentation wird das Auge während der Operation von außen eingedellt, so dass der Chirurg Bereiche erreichen kann, die sonst nicht im Situs liegen. Diese Eindellung verändert das Volumen des Augapfels und somit den Augeninnendruck.

Um dieses Verhalten zu berücksichtigen, wird, wenn es zu einer Eindellung kommt, die Compliance-Funktion  $C(V)$  neu berechnet, so dass als Normalvolumen das um die Eindellung verringerte Volumen verwendet wird. Dem neuen Normalvolumen wird dann der Normaldruck zugewiesen. Danach befindet sich noch mehr Flüssigkeit im virtuellen Auge. Der Druck wird über die Compliance-Funktion angepasst. Durch Ausströmen der Flüssigkeit passt sich der Druck wieder dem Normaldruck an. Bei Verringerung der Indentation verhält es sich umgekehrt.

Der Chirurg erkennt den Überdruck im Auge in der Realität durch ein Pulsieren der Sehnervenscheibe (engl. *optic disc*) im systolischen bzw. diastolischen Rhythmus des Blutdrucks. Der Unterdruck lässt die Augenlinse absinken. Dies führt zu einer unscharfen Darstellung im Mikroskop. Beide Effekte werden in Eyesi berücksichtigt.

## 6.7 Ergebnisse

### 6.7.1 Vorderkammer

Zurzeit liegen wenige frei verfügbare Studien vor, die sich mit der Messung des IOP während der Phakoemulsifikation beschäftigen. Die folgenden Werte beziehen sich auf die Simulation und werden mit Werten aus der Theorie verglichen.

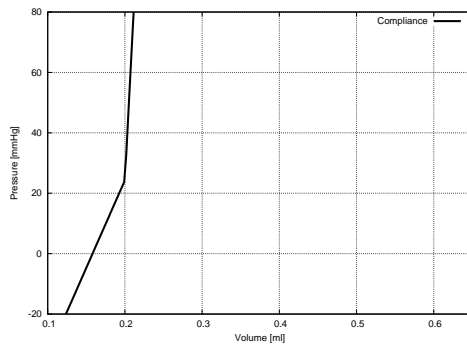
Der Maschinentester eignet sich sehr gut, um das Verhalten der Maschine zur Laufzeit zu beobachten. Für eine bessere Darstellung wurden die Werte in eigenständige Grafiken übertragen. Für die folgenden Messungen wurde im Maschinentester jeweils eine laufzeitabhängige Ereignisabfolge angelegt.

**Auswirkungen der Vorderkammer-Compliance** Das Druckverhalten der simulierten Vorderkammer ist bei dem hier vorgestellten Modell abhängig von der Compliance-Funktion, dem Verhältnis von Normalvolumen zu Druck und den Ober- und Untergrenzen von Volumen und Druck.

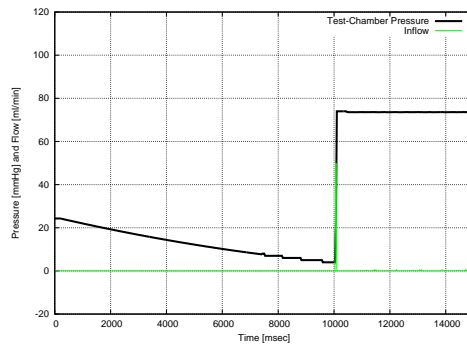
Um den Zusammenhang von Volumen und Druck zu verdeutlichen, wurden Messungen für eine lineare und eine nichtlineare Compliance-Funktion durchgeführt. Das Normalvolumen betrug 0,2 ml und der dazugehörige IOP 24,3 mmHg, die Volumenuntergrenze lag bei 0,14 ml und bei einem IOP -10,5 mmHg bezogen auf den Umgebungsdruck. Für die Volumenobergrenze wurde 0,21 ml, 0,26 ml und 0,6 ml bei gleicher Druckobergrenze von 73,56 mmHg (100 cmH<sub>2</sub>O) verwendet.

In Abbildung 6.9 sind links die Compliance-Funktion und rechts die Auswirkungen auf den Druck in der Vorderkammer dargestellt. Zunächst wurde die Leckage aktiviert, dabei sieht man wie der Druck der Vorderkammer stetig abnimmt. Beim Einsetzen der Irrigation mit einer Flaschenhöhe von 100 cm beginnt der Druck abhängig vom Volumen wieder zu steigen, bis der maximale Druck erreicht wird. Durch die Volumenobergrenze lässt sich die Dauer dieses Prozesses regulieren.

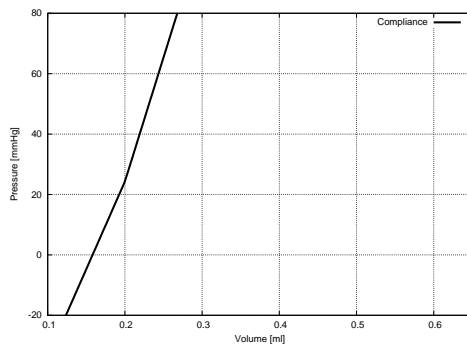
In Abbildung 6.10 sind die gleichen Messungen durchgeführt worden wie für die Vorderkammer mit linearer Compliance-Funktion. Bei der Vorderkammer mit nichtlinearer Compliance fällt der Druck nicht so schnell ab, da die Steigung der verwendeten Funktion geringer ist. Bei aktivierter Irrigation steigt der Druck in der Vorderkammer entsprechend der verwendeten Funktion zunächst schwach und dann stärker. Nähert sich das Volumen der Volumenobergrenze, steigt der Druck schneller als bei der linearen Compliance-Funktion. Erreicht das Volumen die vorgegebene Volumenobergrenze, wird der Druck



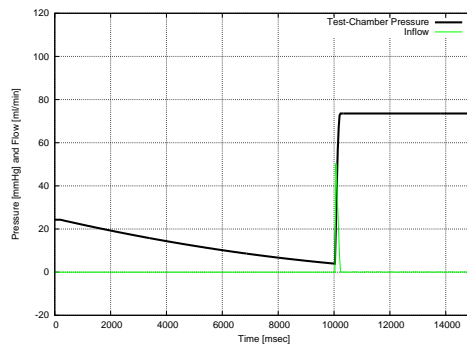
(a) 0,21 ml Compliance



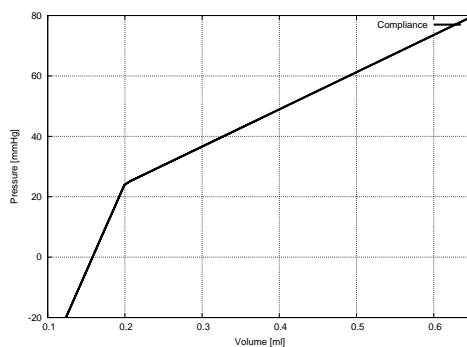
(b) 0,21 ml Volumenobergrenze



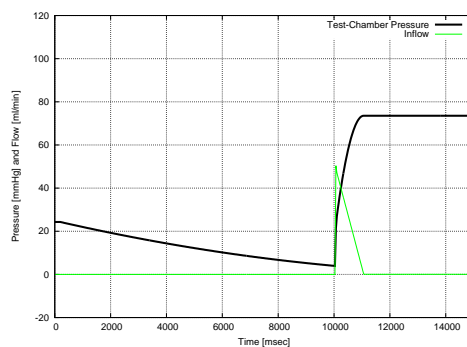
(c) 0,26 ml Compliance



(d) 0,26 ml Volumenobergrenze



(e) 0,6 ml Compliance



(f) 0,6 ml Volumenobergrenze

Abbildung 6.9: Auswirkungen der Volumenobergrenze für die lineare Vorderkammer Compliance-Funktion.



auf den vorgegebenen Maximaldruck begrenzt. Dabei entspricht der Maximaldruck dem größten einstellbaren Irrigationsdruck. Im mittleren Bereich sind die Unterschiede zur linearen Version gering.

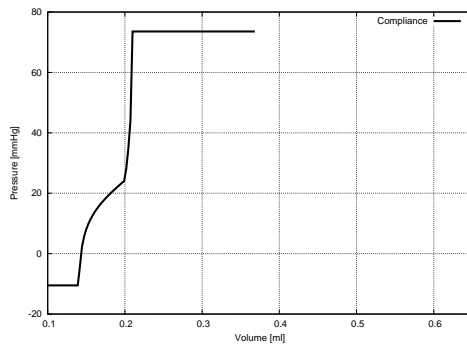
Zurzeit existiert kein weiteres veröffentlichtes Druck- und Volumenmodell der Vorderkammer während einer Echtzeitsimulation der Phakoemulsifikation, deshalb können die berechneten Werte nur eine Vorstellung von den modellierten Vorgängen liefern. Das gesamte Verhalten ergibt sich erst im Zusammenspiel aller Teilkomponenten: der Maschinen-Simulation, das Schlauchsystem und die Vorderkammer.

**Auswirkungen der Maschinen-Simulation und der Aspirationsschlauch-Compliance auf die Vorderkammer** Nachdem die Ergebnisse aller Teilkomponenten präsentiert wurden, sollen nun die Gesamtergebnisse vorgestellt werden. Verglichen werden diese mit den theoretischen Werten, wie sie Seibel in [80] angibt. In Abbildung 6.11 ist dieser theoretische Druckverlauf dargestellt. Bei seinen Betrachtungen geht er von einer Peristalik-Pumpe mit einem Fluss von 30 ml/min, einer Flaschenhöhe von 30 Zoll (ca. 76 cm) und einem Vakuumlimit von 50 mmHg aus.

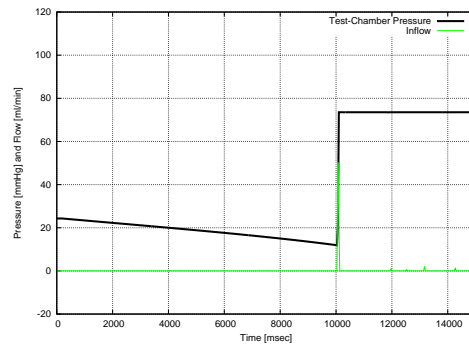
Zuerst wird die Irrigation in Fußpedalstellung **A** aktiviert, der Augeninnendruck steigt auf die eingestellten 30 Zoll Wassersäule. Das Fußpedal wird weiter in den Bereich **B** herabgedrückt, dabei stellt sich ein stationärer Fluss und Druck ein, der um den dynamischen Druck der strömenden Flüssigkeit reduziert ist. Der Augenblick der Instrumentverstopfung ist in Abbildung 6.11 mit „Okklusion“ markiert. Der Druck in der Vorderkammer steigt wieder auf den Wert der Wassersäule an. Bricht die Okklusion auf, setzt der Fluss wieder ein und es kommt zum Surge. Nachdem sich ein Gleichgewicht eingestellt hat, herrscht ein stationärer Fluss und Druck. Abschließend wird das Fußpedal losgelassen, bis es Stellung **A** erreicht hat.

Dieses Szenario wurde mit dem Maschinentester nachgestellt. Die Steuerung erfolgte über die Fußpedalstellung und dem Wert der Okklusion. Der Ablauf entspricht dem von Seibel [80] verwendeten.

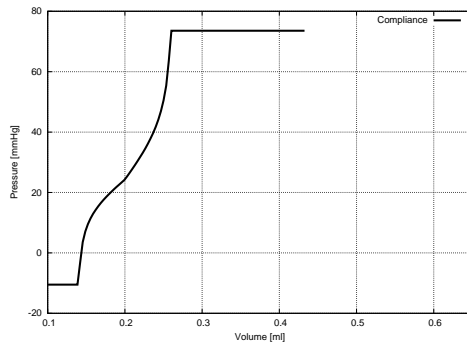
In den Abbildungen 6.12 und 6.13 werden die Druck und Volumenverhältnisse, der Irrigationsfluss (engl. *Inflow*) und der Aspirationsfluss (engl. *Outflow*) für verschiedene  $\tau$ -Werte der Schlauch-Compliance dargestellt. Über den  $\tau$ -Wert lässt sich das Zeitverhalten der Compliance steuern. Das Entladeverhalten ist für die Dauer des Surge und das Aufladeverhalten für die Stärke verantwortlich. Hier wurde für den Aufladewert  $\tau_a = 2$  gewählt und nur der Entladewert  $\tau_e$  variiert. In Abbildung 6.12 wird für die Vorderkammer bzw. die Test-Chamber eine lineare Compliance-Funktion verwendet und in Abbildung 6.13 die nichtlineare artanh-Funktion.



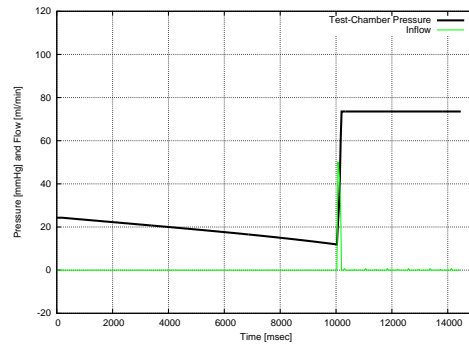
(a) 0,21 ml Compliance



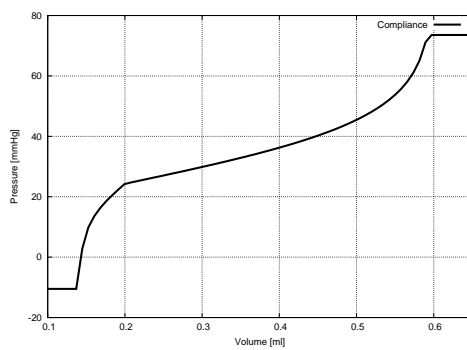
(b) 0,21 ml Volumenobergrenze



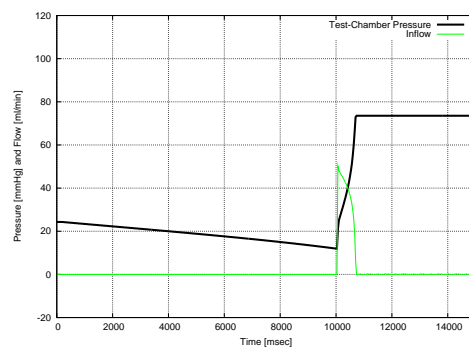
(c) 0,26 ml Compliance



(d) 0,26 ml Volumenobergrenze



(e) 0,6 ml Compliance



(f) 0,6 ml Volumenobergrenze

Abbildung 6.10: Auswirkungen der Volumenobergrenze für die nichtlineare Vorderkammer Compliance-Funktion.

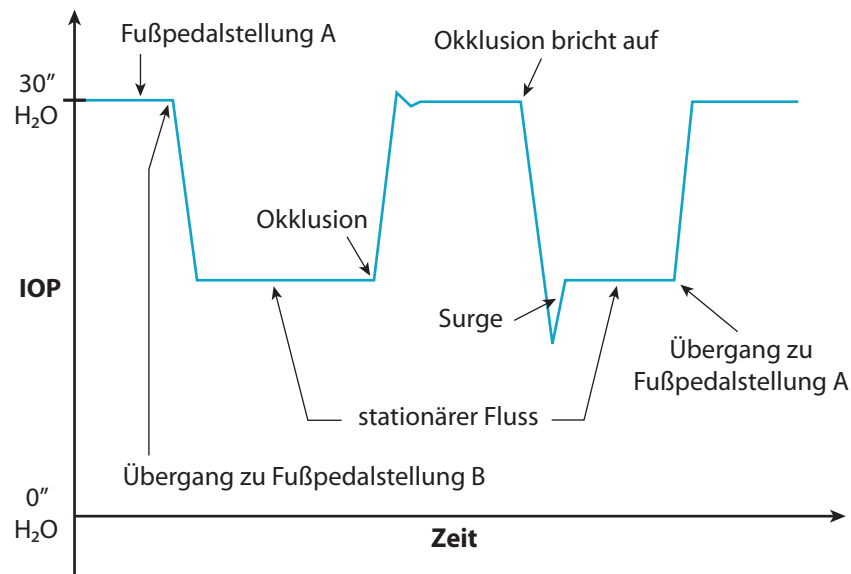


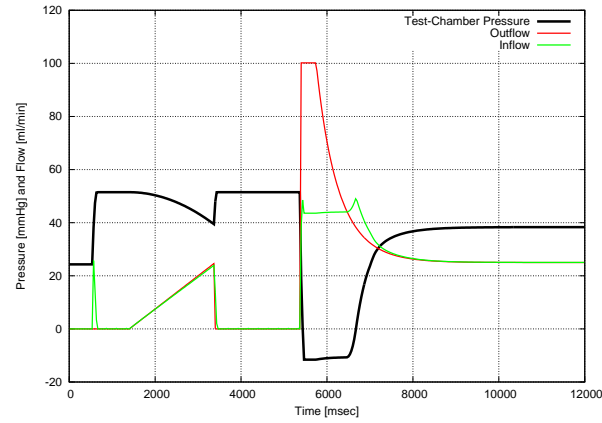
Abbildung 6.11: Der Druckverlauf in der Vorderkammer nach Seibel mit ein Peristaltik-Pumpe und 50 mmHg Vakuumlimit.

Vergleicht man den Kurvenverlauf in den Abbildungen 6.12 und 6.13, stellt man nur geringe Abweichungen fest, die sich aus der Umrechnung vom Vorderkammervolumen zu -druck über die verwendete Compliance-Funktion ergeben.

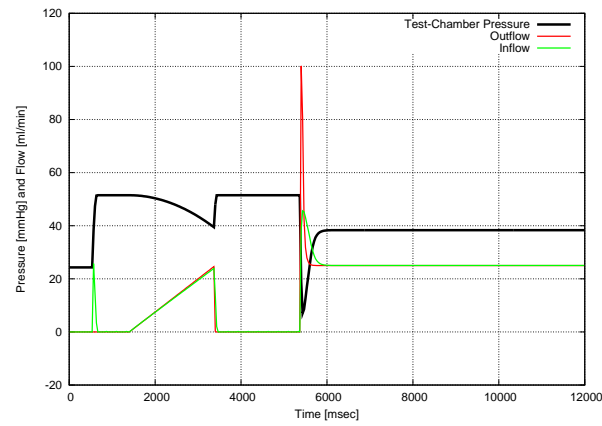
Im theoretischen Verlauf für den IOP (siehe Abbildung 6.11) sind bei der Okklusion und beim Aufbrechen der Okklusion Überschwinger angegeben. Da sich für die lineare Compliance-Funktion stärkere Überschwinger als bei der nichtlinearen Version ergeben, wird in Eyesi die lineare Compliance-Funktion verwendet.

Der im Aspirationsschlauch herrschende Druck (engl. *Aspiration Line Pressure*, kurz ALP), der sich während einer Okklusion aufbaut, ist beim Aufbrechen der Okklusion für den Surge verantwortlich. In dem hier vorgestellten Modell lässt sich die Dauer bis zum Ausgleich des Aspirationsschlauchdrucks über den  $\tau_e$ -Wert einstellen. In Eyesi wird  $\tau_e = 50$  für das Zeitverhalten verwendet.

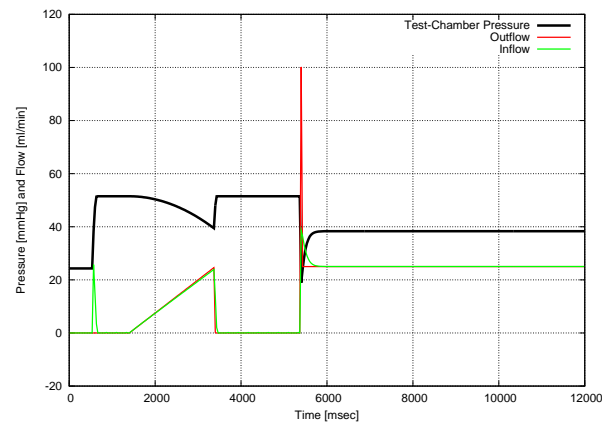
**Surge** Abbildung 6.14 gibt den Surge-Verlauf nach Seibel [80] für unterschiedliche Vakuumlimits an. Es wird eine Peristaltik-Pumpe mit einem Fluss von 30 ml/min, einer Flaschenhöhe von 30 Zoll und Vakuumlimits von 100, 160 und 250 mmHg verwendet. Bei einem Surge bei Vakuumlimit 160 bzw. 250 mmHg gibt Seibel einen Kammerkollaps mit einem IOP unter 0 (bezogen



(a)  $\tau_e = 2$



(b)  $\tau_e = 20$



(c)  $\tau_e = 50$

Abbildung 6.12: Surge Szenario für eine Test-Chamber mit linearer Compliance-Funktion und verschiedene  $\tau$ -Werte für den Aspirationsschlauch.

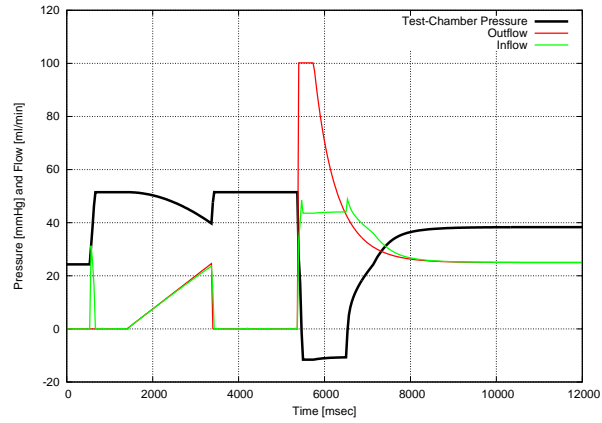
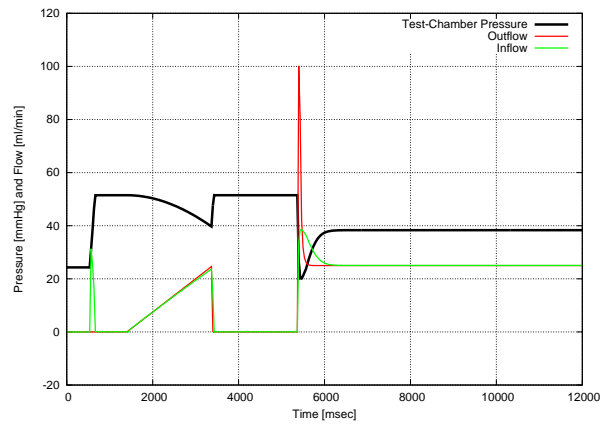
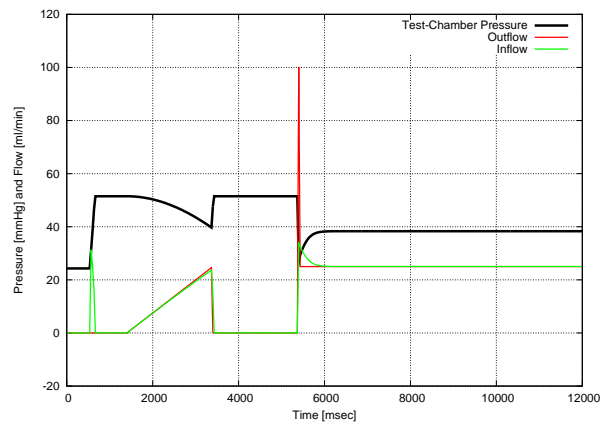
(a)  $\tau_e = 2$ (b)  $\tau_e = 20$ (c)  $\tau_e = 50$ 

Abbildung 6.13: Surge Szenario für eine Test-Chamber mit nichtlinearer Compliance-Funktion und verschiedene  $\tau$ -Werte für den Aspirationsschlauch.

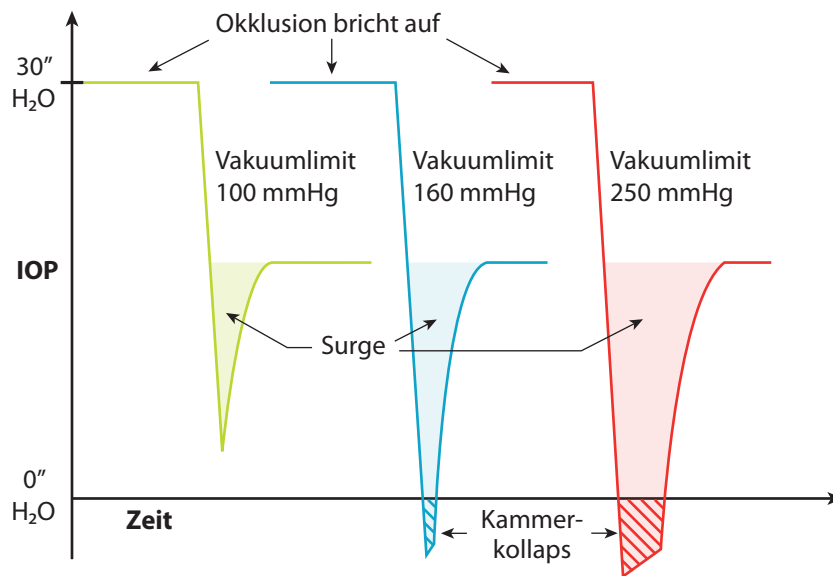


Abbildung 6.14: Der Surge in der Vorderkammer nach Seibel für verschiedene Vakuumlimits.

auf den Umgebungsdruck) an.

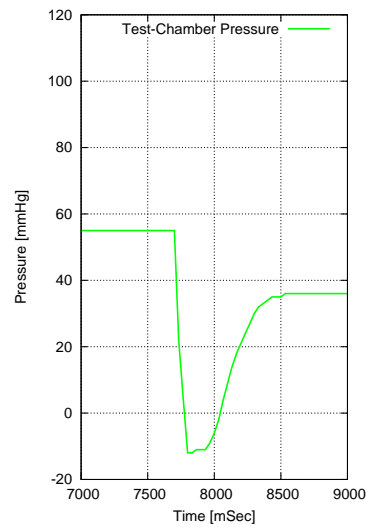
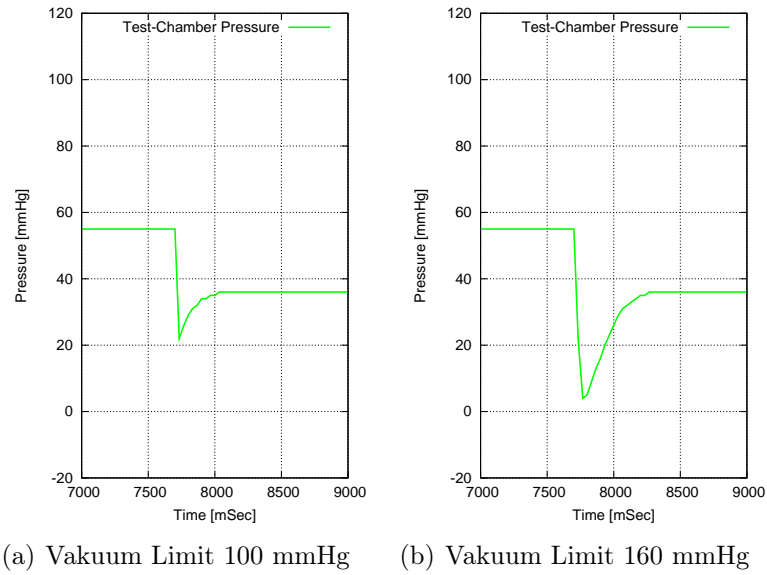
Die Messungen, die in Abbildung 6.15 dargestellt sind, orientieren sich an den Betrachtungen von Seibel [80]. Sie wurden mit einer simulierten Periskaltik-Pumpe mit einem Fluss von 30 ml/min, einer Irrigationshöhe von 75 cm und unterschiedlichen Vakuumlimits durchgeführt. Die Stärke des Surge lässt sich in dem hier verwendeten Modell über das Verhältnis von Vorderkammervolumen zu Vorderkammerdruck und über die Compliance-Funktion des Aspirationsschlauches einstellen.

Vergleicht man die Werte mit denen aus [80], stellt man fest, dass der Surge für niedrige Vakuumwerte bei der simulierten Maschine geringer ist. Für höhere Vakuumwerte nähert sich die Kurve deutlich den Werten aus der Theorie.

Die durchgeführten Messungen zeigen, dass das entwickelte Modell zu einem plausiblen Vorderkammerverhalten während der Simulation führt. Dies bestätigt auch der Vergleich mit theoretischen Betrachtungen, wie sie Seibel in [80] anstellt.

### 6.7.2 Hinterer Augenabschnitt

Das derzeitige Modell erlaubt es, den Druck und das Volumen des Auges durch Eindellungen (engl. *Indentation*) anzupassen. Der Ausgleich des



(c) Vakuum Limit 250 mmHg

Abbildung 6.15: Der Druck in der Test-Chamber mit linearer Compliance-Funktion für verschiedene Vakuumobergrenzen entsprechend der theoretischen Betrachtungen von Seibel [80].

Druckanstiegs durch die Eindellung erfolgt über die Leckage und die Aspiration.

In Abbildung 6.16 ist das Verhalten während einer Eindellung mit anschließender Leckage dargestellt. Der Ablauf der automatischen Steuerung war: Einschalten der Irrigation, Erhöhen der Eindellung auf 100%, Erhöhen der Leckage auf 100%, Reduzieren der Leckage auf 0%, Reduzieren der Eindellung auf 0%. In Abbildung 6.16(a) erfolgt keine Aspiration. In Abbildung 6.16(b) erfolgt die Aspiration mit einer Venturi-Pumpe bei 250 mmHg Vakuumlimit.

Unter dem Gesichtspunkt, dass es sich hier um extreme Werte für die Eindellung und Leckage handelt, erkennt man wie sich die Indentation auf den Augeninnendruck auswirkt. Über die Leckage reduziert sich das Augvolumen, bis das um die Eindellung verringerte Augvolumen erreicht wurde. Der Augeninnendruck hat sich dem neuen Normalvolumen angepasst. Der Unterdruck, der durch das Lösen der Eindellung erzeugt wird, kompensiert sich sehr schnell durch die Irrigation.

Bei aktivierter Pumpe erfolgt der Ausgleich des durch die Eindellung entstehenden Drucks wesentlich schneller. Der Ausgleich des Unterdrucks durch Lösen der Eindellung erfolgt in diesem Beispiel jedoch langsamer, da der durch Irrigation erzeugt Zufluss im Verhältnis zum Abfluss zu gering ist.

### 6.7.3 Parameter für Eyesi

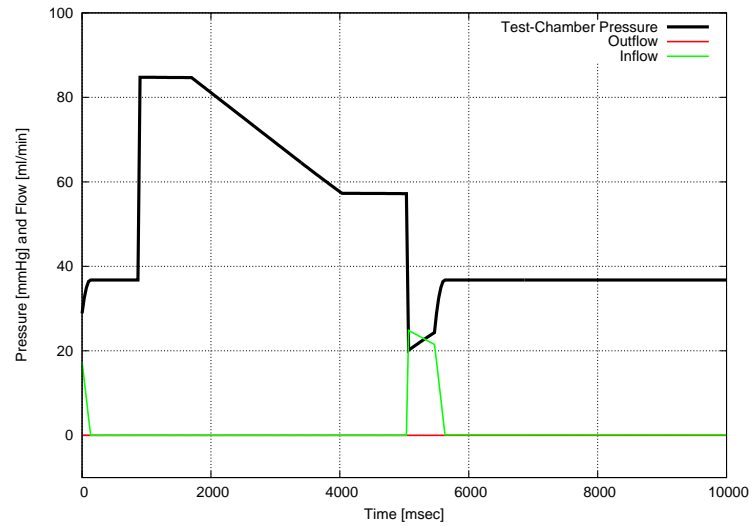
Die Modelle der Maschine und der Vorderkammer bzw. des hinteren Augenabschnitts verfügen über eine Reihe von Parametern. Für Simulationen am vorderen Augenabschnitt ist zurzeit nur ein Standard-Phakohandstück grafisch modelliert. Die Modellparameter wurden so gewählt, dass die Auswirkungen der Schläuche und der Vorderkammer zu einem moderaten Surge-Verhalten führen. Auch wenn dieses bei neueren Maschinen immer geringer wird. Sollte der angehende Chirurg mit diesem zurecht kommen und ist die Simulation, im Vergleich zur Realität zu leicht, kann das zur Folge haben, dass der Lernende zur sorglos wird.

Im hinteren Augenabschnitt spielt der Surge keine Rolle, die Parameter wurden so gewählt, dass sich Druck und Volumen für die entsprechende Flaschenhöhe, Aspirationseinstellungen und Schnittrate des Vitrektors plausibel verhalten.

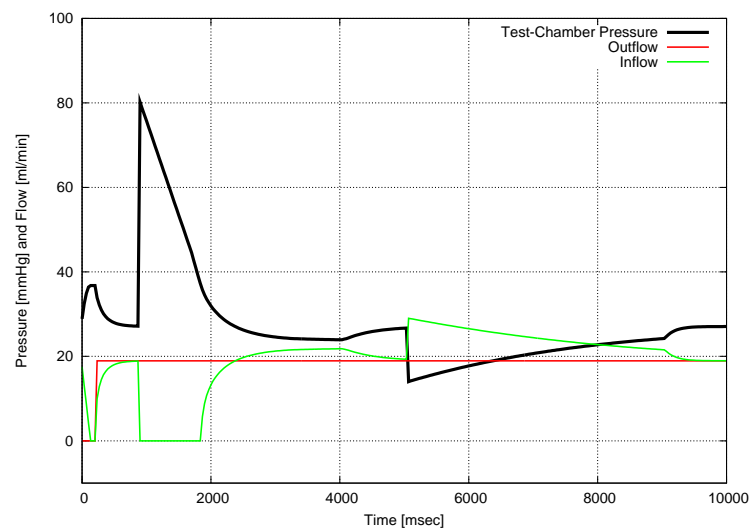
## 6.8 Diskussion

Für die Simulation der Phakoemulsifikation müssen viele Bereiche abgedeckt werden. Für jeden einzelnen Bereich der Operation wurde ein Modell ent-





(a) Ohne aktivierte Pumpe



(b) Mit aktivierter Pumpe

Abbildung 6.16: Abhängigkeit von Zufluss, Abfluss und Druck während einer Indentation.

wickelt. Dies sind Modelle für die mechanischen Auswirkungen, das Druck- und Volumenverhalten, das Verhalten der Objekte in der Flüssigkeit, das Verflüssigen der Objekte durch Ultraschall und das Absaugen durch Vakuum. Diese Modelle berücksichtigen mechanische und maschinelle Eingaben in die Simulation.

Handlungsbedarf besteht bei den Flüssigkeitsmodellen. Zurzeit werden diese getrennt voneinander behandelt, so dass sich Änderungen am Maschinenmodell unterschiedlich auf das Druck- und Volumenverhalten der Vorderkammer und auf die Bewegung der Linsenfragmente auswirken können. Dies könnte durch ein kombiniertes Modell verhindert werden.

Darüber hinaus wurde das Druck- und Volumenmodell auf den hinteren Augenabschnitt übertragen und angepasst. Es kommt in allen vitreoretinalen Trainingsmodulen zum Einsatz.

## Software-Architektur

In diesem Kapitel wird die Integration der OMS und der Flüssigkeitssimulation in das Eyesi-System behandelt. Dabei wird das System nur ausschnittsweise beschrieben. Die komplette Implementierung erstreckt sich über weite Teile der vorhandenen Eyesi-Software. Aktuell besteht sie aus den Maschinenbasisklassen für die Ablaufsteuerung und für die Kommunikation über Interfaces mit Eyesi und den Maschinen-GUIs, einer spezialisierten OMS Implementierung, der *EsORMachine* und der Anbindung an vier Maschinen-GUIs. Bei zweien handelt es sich um Nachbauten realer Maschinen-GUIs, bei einer um die Anbindung der existierenden GUI wie sie in der realen Maschine verwendet wird und bei einer um eine Eigenentwicklung, der *VRmagicMachineGUI*. Verwaltet und angesteuert werden die Maschinen von Eyesi im *EsORMachineManager*.

Des Weiteren wurde der Maschinentester - eine Testumgebung für die OMS - mit Anbindung der Maschinen-GUI und die Druck- und Volumen-Simulation der Vorderkammer bzw. des hinteren Augenabschnitts entwickelt.

### 7.1 Stand der Technik

Aktuelle VR-Trainingssysteme bieten die Möglichkeit, die simulierte Szene direkt zu manipulieren. Die Bewegungen, die der Chirurg ausführt, werden über ein optisches oder mechanisches Tracking in die virtuelle Realität übertragen. Dort bilden sie die Grundlage für die Manipulation der virtuellen Szene, die wiederum dem Chirurg grafisch sowie zum Teil auch haptisch dargestellt wird.

Die Integration von Operationsmaschinen in dieses Szenario stellt eine Metaebene in diesem Konzept dar. Der Chirurg manipuliert die virtuelle Szene nicht mehr direkt, sondern bedient eine Maschine, deren Einstellungen eine

spezielle Auswirkung auf die Szene haben. Ebenso müssen sich Vorgänge in der simulierten Szene auf die Maschine auswirken.

Die Entwicklung eines solchen Konzeptes erfordert die Betrachtung der Software-Architektur von Eyesi und die von realen Operationsmaschinen.

**Eyesi** Die Software des Eyesi Trainingssimulators basiert auf den VRM-Libs [99] und [32]. Die VRM-Libs sind Bibliotheken, die für die Entwicklungen vor VR-Simulatoren entwickelt wurden. Sie beinhalten Algorithmen zur Kollisionserkennung, zur Simulation, wie sie teilweise in *Kapitel 3 - Grundlagen der Simulation* beschrieben wurden, zur Ein- und Ausgabe (I/O) und viele weitere Grundaufgaben. Neben der Simulationsumgebung beinhaltet die Eyesi-Software weitreichende Verwaltungsmöglichkeiten.

Zu Beginn der Entwicklung des Simulators lag das Hauptaugenmerk auf einer realistischen physikalischen Simulation und grafischen Darstellung der simulierten Augenumgebung [76], [99] und [32]. Ein zweiter Hauptbestandteil war die Entwicklung einer realistischen Hardwarebenutzerschnittstelle. Mit fortschreitender Entwicklung wurde das Gerät mit einem Fußpedal zur Steuerung aktiver Instrumenteigenschaften ausgestattet. Die von der Hardware zur Verfügung gestellten Werte wurden in der Simulation, meist in der Instrumentklasse selbst, verwendet, um die Instrumenteigenschaften zu berechnen.

Spezielle Ausprägungen der Operationsmaschine und deren Auswirkungen auf die Simulation wurden nicht berücksichtigt. Weiterhin gab es keine definierten Schnittstellen für den Umgang mit den Maschinenwerten bzw. den Fußpedalen. Diese wurden für jedes Fußpedal und Instrument immer wieder neu implementiert.

**Reale Operationsmaschinen** In den letzten Jahren wurde die Entwicklung realer Operationsmaschinen immer weiter voran getrieben. Seit einiger Zeit basieren die Maschinen auf PC- und mikroprozessorgesteuerten Systemen, deren Bedienung über einen an der Maschine befestigten Touchscreen erfolgt. Über die GUI erhält der Benutzer Zugriff auf die Maschine. Dabei kann zwischen Verwaltungsaufgaben und Maschinenkonfiguration unterschieden werden. Zu den Verwaltungsaufgaben zählen zum Beispiel die Angabe von Benutzernamen oder die Protokollierung der durchgeführten Operationen.

Zur Maschinenkonfiguration zählen die Festlegungen der Maximalwerte, der Fußpedaleinstellungen und der Pumpe. Die interne Regelung der verwendeten Komponenten erfolgt in der Software, dafür werden entsprechende Sensorwerte berücksichtigt.

Die Entwicklung von Operationsmaschinen mit Touchscreen erleichtert die Integration in die virtuelle Realität. Die notwendigen Werte können darüber direkt eingegeben werden. Somit ist es nicht nötig, spezielle Hardware zu entwickeln, die die Werte an einer realen Operationsmaschine ausliest und in die virtuelle Realität überträgt.

Im Gegensatz zur Realität kann nicht einfach eine neue Maschine an das Handstück angeschlossen werden. Die Maschineneigenschaften müssen in der virtuellen Realität abgebildet werden. Die Grundlage für diese Berechnungen der Maschinen-Simulation wurden in Kapitel 5 - *Physik der Phakoemulsifikation - Die Operationsmaschine* behandelt. Nun sollen die entwickelten Software-Anpassung näher erläutert werden.

## 7.2 Software-Erweiterungen

Ziel der Software-Erweiterungen ist, Eyesi um verschiedene Operationsmaschinen und Maschinen-GUIs zu erweitern, um die Realitätsnähe des Systems zu erhöhen. Dafür wurde das Modell einer Operationsmaschine und verschiedene Maschinen-GUIs implementiert. Für die Integration der Operationsmaschine und der Maschinen-GUI in das vorhandene System wurden geeignete Schnittstellen geschaffen, damit die unterschiedlichen Teile miteinander kommunizieren können.

Die Software lässt sich grob in drei Teile aufspalten:

- Eyesi,
- Operationsmaschinen-Simulation (OMS),
- Maschine GUI.

Dabei ermöglicht Eyesi den Zugriff und die Verwaltung der gesamten Hardware des Operationssimulators. Die OMS berechnet die für die virtuelle Realität notwendigen laufzeitabhängigen Werte. Diese werden Eyesi und der Maschinen-GUI zur Verfügung gestellt.

Die Maschinen-GUI stellt die spezielle Ausprägung der Maschine dar. Über sie ist es möglich, die Operationsmaschine zu konfigurieren. Außerdem beinhaltet sie Verwaltungsaufgaben der Maschine, wie Arzt- und Operationsauswahl.

Wie in [99] wurde auch hier eine strikte Trennung von I/O und Verarbeitung der Daten eingehalten. Dies ermöglicht den einfachen Austausch der Benutzeroberflächen. Des Weiteren wurde darauf geachtet, dass nur die relevanten Werte in die virtuelle Realität übertragen werden, um eine leicht

erweiterbares und flexibles System zu erhalten und die Abhängigkeiten zu minimieren.

Diese Architektur ermöglicht die Maschinen-GUI komplett von Eyesi zu entkoppeln, so dass sie in einem eigenen Thread lauffähig ist und die Parallelität neuer Prozessorgenerationen besser genutzt werden kann.

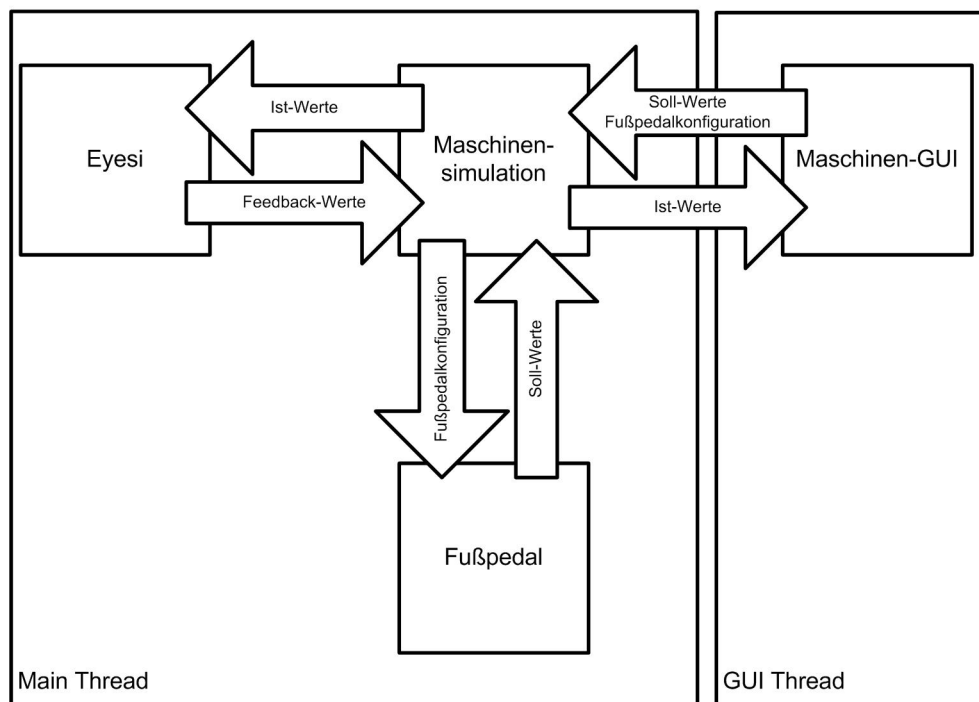


Abbildung 7.1: Die Datenabhängigkeiten zwischen Eyesi, MaschinentSimulation, Fußpedal und Maschinen-GUI.

**Datenabhängigkeiten** In Abbildung 7.1 sind die Datenabhängigkeiten zwischen Eyesi, MaschinentSimulation, Fußpedal und Maschinen-GUI dargestellt. Die Operationsmaschinen-Simulation nimmt eine zentrale Stelle ein. Dort werden alle Werte verarbeitet und Eyesi sowie der Maschinen-GUI bereitgestellt. Eine Besonderheit stellt die Konfiguration des Fußpedals dar. Das Maschinenfußpedal ist ein VR-Eingabegerät, das in Eyesi verwaltet wird. Dadurch wird gewährleistet, dass alle für die VR benötigten Daten während des Trainings aufgezeichnet werden. Ebenso wird die Trennung zwischen I/O und VR aufrechterhalten. Bei den meisten realen Maschinen kann das Fußpedal

frei konfiguriert werden. Um dies zu ermöglichen, werden die Konfigurationsdaten von der GUI durch die Maschine an das Fußpedal weitergereicht. Die Werte, die Eyesi vom Fußpedal liefert, werden im *Footpedalmapping* weiterverarbeitet, so dass sie direkt in der OMS verwendet werden können.

**Footpedalmapping** Je nach Art und aktueller Stellung des Fußpedals können verschiedene Eigenschaften der Maschine aktiviert werden. Das Footpedalmapping wurde deshalb so allgemein wie möglich gestaltet. Damit über das Maschinenfußpedal verschiedene Eigenschaften der Maschine manipuliert werden können, verwaltet das Footpedalmapping eine Reihe von Einträgen. Dabei wird zwischen Einträgen für analoge und digitale Werte unterschieden.

Exemplarisch wird hier der analoge Eintrag näher behandelt.

```
1 struct tAnalogFootpedalMappingEntry
2 {
3     std::string name;
4     EsID::eORMachinePropID axbutID;
5     EsID::eORMachinePropID targetID;
6     float startValue;
7     float stopValue;
8     float belowValue;
9     float minValue;
10    float maxValue;
11    float aboveValue;
12    float duration;
13    float resetTime;
14    float activationTime;
15 };
```

Quellcode 7.1: Die Struktur eines analogen Footpedalmapping-Eintrags

Die Struktur eines analogen Footpedalmapping-Eintrags ist in Quellcode 7.1 dargestellt. Dabei gelten folgende Zuordnungen:

- name - der Name des Eintrags,
- axbutID - der Bezeichner für die Achse zu dem der Eintrag gehört,
- targetID - der Bezeichner des zu ändernden Zielwertes,
- startValue - der Startnormwert, ab dem der Zielwert aktiviert wird,
- stopValue - der Stoppnormwert, ab dem der Zielwert seinen Maximalwert erreicht hat,
- belowValue - der Wert, der gesetzt wird, wenn der Startnormwert unterschritten wird,
- minValue - der Startwert des Mappings,

- `maxValue` - der Stoppwert des Mappings,
- `aboveValue` - der Wert, der gesetzt wird, wenn der Stoppnormwert überschritten wird,
- `duration` - die Dauer bis zum automatischen Zurücksetzen des Wertes,
- `resetTime` - die Dauer bis der Wert erneut angenommen werden kann,
- `activationTime` - die Zeit, die sich das Fußpedal im aktiven Bereich befindet.

Über den „`minValue`“ und „`maxValue`“ wird mit Hilfe des Footpedalmappings der Soll-Wert der Maschine konfiguriert. Dabei ist der Linear- und Fix-Betrieb berücksichtigt worden. Der Fix-Wert ergibt sich implizit, wenn der „`minValue`“ gleich „`maxValue`“ ist. Dies erleichtert die Berechnungen der Maschinen-Simulation. Dort muss nicht zwischen Linear- und Fix-Betrieb unterschieden werden.

Im Footpedalmapping werden alle über das Fußpedal manipulierbaren Einträge verwaltet und verarbeitet. Bei einer Betätigung des Fußpedals ändert sich der Fußpedalnornwert. Diese Änderung wird im Footpedalmapping ausgewertet und der Zielwert wird über die „`targetID`“ gesetzt.

Die Konfiguration des Footpedalmappings findet beim Programmstart statt, dabei werden die entsprechenden Werte geladen. Dies geschieht entweder über eine Datei, in der Standardwerte abgespeichert sind, oder benutzerabhängig aus der Eyesi-Datenbank. Die Änderung des Footpedalmappings führt der Benutzer über die Maschinen-GUI durch. Somit benötigt die Maschinen-GUI Zugriff auf das Footpedalmapping. Dies wird durch die später vorgestellte Kommunikationsstruktur erreicht.

Für unterschiedliche Fußpedaltypen stehen hierbei unterschiedliche Konfigurationen zur Verfügung, die in Eyesi verwaltet werden. Durch dieses allgemeine Konzept der Fußpedalverwaltung bestehen an anderen Stellen keine Abhängigkeiten von dem eingesetzten Fußpedaltyp und dessen Konfiguration wie dies zuvor der Fall war.

**Kommunikation** Kommunikationsbedarf besteht an mehreren Stellen der vorgestellten Architektur. Die beiden Hauptbereiche sind die Kommunikation zwischen OMS und Maschinen-GUI sowie zwischen OMS und Eyesi.

Die strikte Trennung von virtueller Realität und I/O ermöglicht es, die Maschinen-GUI in einem eigenen Thread zu starten. Die OMS befindet sich im Main-Thread und läuft synchron mit der von Eyesi erzeugten virtuellen Realität. Dies macht eine Kommunikation über Thread-Grenzen nötig. Um den Synchronisationsaufwand gering zu halten, wird ein Doppel-Puffer-System



verwendet. Jeder Thread trägt seine Informationen in den entsprechenden Puffer ein. Dieser wird dann zu einem bestimmten Zeitpunkt im Prozessablauf des anderen Threads abgearbeitet und entsprechend abgespeichert.

Über diese Kommunikationsstruktur stehen Kommandos für die allgemeine GUI Steuerung, die Werteübertragung, die Fußpedalkonfiguration, den Trainingsmodulstatus und die Instrumente zur Verfügung.

Für die Kommunikation zwischen OMS und Eyesi ist solch ein Konstrukt überflüssig. Eyesi benötigt nur lesenden Zugriff auf die Simulationswerte an der Instrumentspitze, dafür wurde eine Trennung von Soll- und Ist-Werten durchgeführt. Dabei dürfen die Ist-Werte nur innerhalb der Maschine gesetzt werden. Es wird ein ID-basiertes System benutzt, bei dem jede Variable mit einer bestimmten ID verknüpft wird. Zugriff auf die Werte der Variable erhält man nur über diese ID.

### 7.2.1 Maschinen-Simulation - ORMachine

Die OMS stellt das Bindeglied zwischen den Benutzereingaben auf der Maschinen-GUI, dem Fußpedal und den Auswirkungen in Eyesi dar. Dies wird durch die Unterteilung in Soll- und Ist-Werte erreicht. Die Soll-Werte ergeben sich aus der Maschinenkonfiguration und der Stellung des Fußpedals. Die Ist-Werte werden durch das OMS-Modell für die virtuelle Realität berechnet. Werden zur Berechnung der Ist-Werte Informationen von der virtuellen Realität benötigt, so werden diese als Feedback-Werte bezeichnet. Die Okklusion der Instrumentspitze ist zum Beispiel solch ein Wert.

Es werden zwei Arten von Ist-Werten berechnet: Zum einen die Werte, die an der Instrumentspitze herrschen und zum anderen die, die an der Maschine bzw. an Sensoren der Maschine herrschen.

Die Berechnungen der Ist- und Soll-Werte für die virtuelle Realität erfolgt dem in *Kapitel 5 - Physik der Phakoemulsifikation - Die Operationsmaschine* vorgestellten Modell. Die OMS läuft synchron mit der Eyesi-Simulation und benutzt deren Zeitschritt.

**Ablauf der Simulation** Die Berechnungen in der *Process*-Routine der OMS sind in einzelne Funktionskomponenten unterteilt und laufen sequentiell ab. Es existieren Funktionskomponenten für die Irrigation, die Aspiration, den Ultraschall und die Vitrektomie. Dabei stehen, je nach Trainingsmodul, nicht alle Methoden zur Verfügung. In den einzelnen Methoden für die Funktionskomponenten erfolgen die Berechnungen der Ist-Werte.

Um die Immersion weiter zu verstärken, wurden ebenfalls die Geräuschausgaben der realen Maschine nachgebildet. Der Operateur erhält so das von der

realen Operation gewohnte akustische Feedback. Die Geräusche werden am Ende des aktuellen Simulationsdurchlaufs auf die aktuellen Werte angepasst.

Nach jedem Simualtionsschritt liegen die aktuellen Werte für Eyesi und für die Maschinen-GUI vor. Durch die Unterteilung in Funktionskomponeten, können gezielt einzelne Methoden in speziellen OMS überschrieben werden. Die Maschinen-GUI kann von der OMS entkoppelt werden und asynchron in einem Thread ausgeführt werden. Die Werte werden gepuffert, damit es bei Anfragen nicht zu Blockierungen kommt. Über Interfaces besteht von beiden Seiten Zugriff auf die berechneten Werte. Ein direkter Zugriff von Eyesi aus auf die Maschinen-GUI und umgekehrt ist nicht möglich.

### 7.2.2 Eyesi

Die Erweiterungen der Eyesi Simulationssoftware beinhalten zwei Bereiche: Zum einen die Verwaltung virtueller Operationsmaschinen und zum anderen die Erweiterungen des virtuellen Auges.

Die virtuellen Operationsmaschinen werden im *EsORMachineManager* verwaltet. Über diese Klasse ist es möglich, die aktuell aktive Maschine und die Ist-Werte der Maschinen-Simulation abzufragen. Die in der OMS berechneten Werte stehen somit in Eyesi zur Verfügung. Dort können sie für weitere Berechnungen genutzt werden.

Sie werden genutzt, um die Druck- und Volumenverhältnisse in der Vorderkammer bzw. im Auge zu berechnen. Dies geschieht trainingsmodulübergreifend in der *EsTrainingModuleBase* (siehe Abbildung 7.2). Dort sind grundlegenden Abläufe eines Trainingsmoduls festgelegt.

Die Druck- und Volumenberechnung erfolgen in der Basisklasse *EsHydrodynamics*. Die für die Berechnungen notwendigen Parameter werden beim Trainingsmodulstart geladen. Die Implementierung richtet sich nach dem Modell aus *Kapitel 6 - Gewebeinteraktion mit dem Instrumentarium*. Die abgeleiteten Klassen beinhalten spezielle Berechnungen für das entsprechende Augensegment. Es wird zwischen vorderem und hinterem Abschnitt unterschieden und die Simulation mit den entsprechenden Werten initialisiert.

Der Ablauf der Flüssigkeitssimulation erfolgt in drei Schritten:

- Vorverarbeitung,
- Simulation und
- Nachbearbeitung.

Im Vorverarbeitungsschritt werden in den abgeleiteten Klassen *EsCataractHydrodynamics* und *EsVitreoretinalHydrodynamics* die für die Simulation

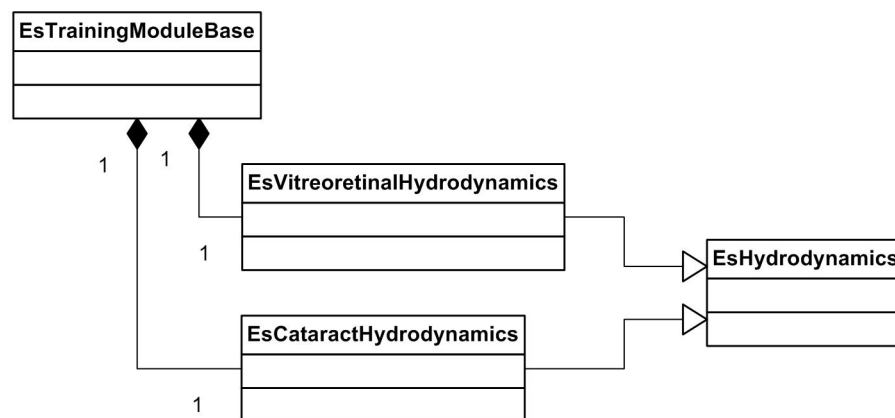


Abbildung 7.2: Der Aufbau und Einsatz der *EsHydrodynamics* Klassen in Eyesi.

notwendigen Werte ermittelt und an die *EsHydrodynamics* übertragen. Dies beinhaltet die Maschinenwerte, abhängig vom aktuellen Instrument und die durch die Instrumentstellung entstehende Leckage. In *EsVitreoretinalHydrodynamics* wird das Volumen einer möglichen Eindellung übermittelt.

In der Basisklasse werden damit die aktuellen Druck- und Volumenverhältnisse für das aktuelle Augsegment berechnet.

In der Nachbearbeitung erfolgen die Anpassungen der virtuellen Szene für den jeweiligen Augabschnitt. Im vorderen Abschnitt ist dies die Abflachung der Cornea bei zu geringem Vorderkammervolumen. Im hinteren Abschnitt führt zu geringer Druck zu einer Unschärfe der Szene und zu hoher Druck zum sichtbaren Pulsieren der Sehnervenscheibe entsprechend des Blutdrucks. Dies wird im *Kapitel 8 - Trainingsmodule* aufgegriffen.

In den Interaktionsklassen und den Trainingsmodulen werden die OMS Werte benutzt, um instrumentspezifische Eigenschaften zu modellieren. So lassen sich mit den Aspirationswerten Fluss und Vakuum, die Bewegungen der simulierten Objekte und mit dem Vakuum die Haltekräfte modellieren.

Die aktiven Instrumenteigenschaften wie Ultraschallenergie oder Schnittrate fließen in das jeweilige Modell zur Gewebedestruktion ein.

Für die OMS müssen von Eyesi die Feedback-Werte für den Verstopfungsgrad gesetzt werden. Dies muss in der zum aktuellen Trainingsmodul gehörenden Interaktion geschehen.

### 7.2.3 Maschinen-GUI

Die Maschinen-GUI dient neben Steuerung der Maschine zur Visualisierung aktueller Werte. Diese können von der OMS abgefragt werden. Durch die

Trennung von GUI und OMS ist es möglich, die Maschine über verschiedene Oberflächen zu steuern. In Abbildung 7.3 ist die VRmagic Maschinen-GUI zu sehen. Die VRmagic Maschinen-GUI ist ein unabhängiger Versuch der Implementierung einer Benutzeroberfläche für die Operationsmaschine. Sie verfügt über zwei Bedienoberflächen: eine für die Phakoemulsifikation (wie in Abbildung 7.3 zu sehen ist) und eine für die Vitrektomie. Der Benutzer kann dort, wie an einer realen Maschine, alle für die Simulation benötigten Werte einstellen.

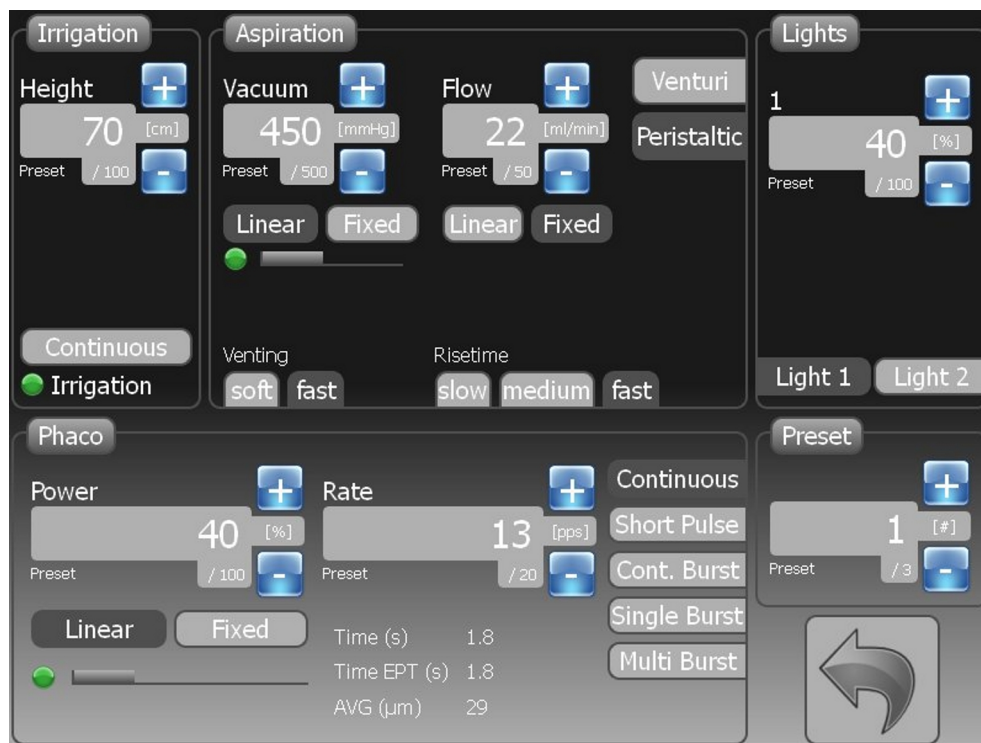


Abbildung 7.3: Die VRmagic Maschinen-GUI für die Phakoemulsifikation.

In der VRmagic Maschinen-GUI sind die einzustellenden Werte nach ihren Eigenschaften gruppiert. Links oben unter „Irrigation“ kann die Flaschenhöhe eingestellt werden. In der Mitte unter „Aspiration“ kann die Pumpe gewählt und konfiguriert werden. Links unten unter „Phaco“ lassen sich die Ultraschalleinstellungen wählen und konfigurieren. Unter „Lights“ kann die Intensität der Lichtquelle angepasst werden und unter „Preset“ kann zwischen verschiedenen Voreinstellungen gewählt werden.

Die VRmagic Maschinen-GUI wird mit Eyesi ausgeliefert. Es wurden schon weitere Benutzeroberflächen für die OMS in Eyesi implementiert. Dies sind

die Alcon Accurus, die AMO Gemini und die Geuder Megatron S3.

## 7.3 Maschinentester

Um die komplexen Zusammenhänge von Fluss, Vakuum, Pumpe, Schläuchen, Druck und Volumen in der Vorderkammer bzw. in dem hinteren Augenabschnitt genauer untersuchen zu können, wurde der Maschinentester entwickelt. Dieser erlaubt das Testen und Visualisieren aller in der virtuellen Realität benötigten Werte. Er umfasst eine komplette Nachbildung der Eingabegeräte, wie Fußpedale und Instrumente, die mit Eyesi verwendet werden können. Über eine *Test-Chamber*, die auf der Hydrodynamik-Simulation, wie sie auch in Eyesi verwendet wird, beruht, lassen sich die zeitlichen Abläufe für Druck und Volumen im virtuellen Auge berechnen und visualisieren.

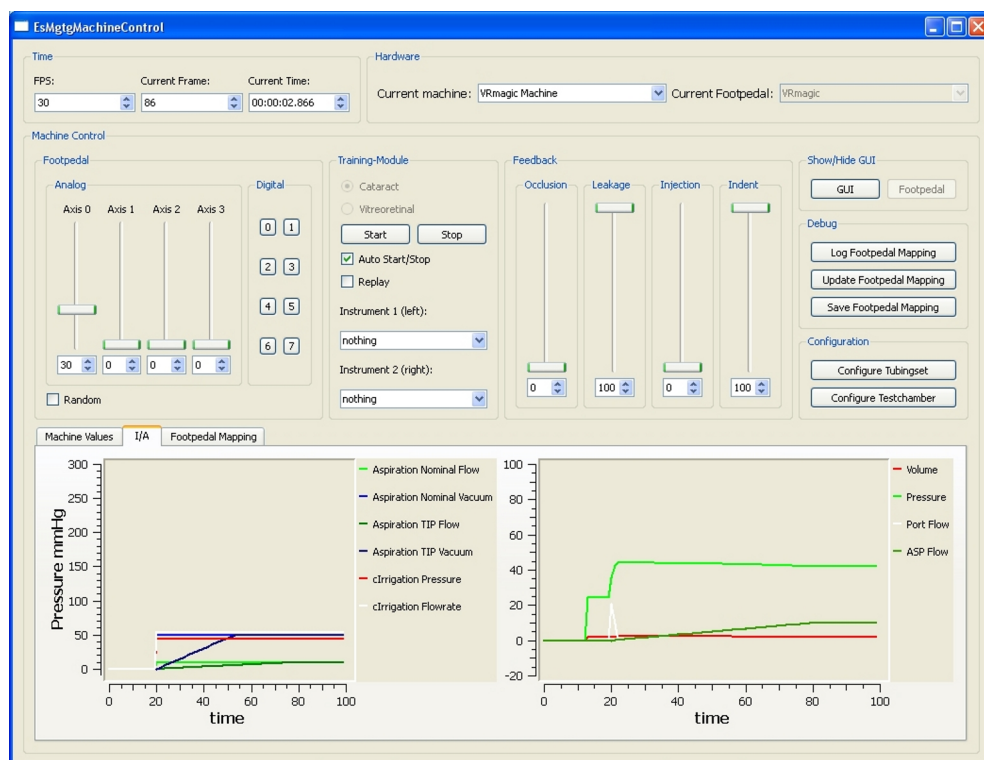


Abbildung 7.4: Der Hauptbildschirm des Maschinentesters mit allen wichtigen Steuerelementen (z. B. den Fußpedal-, den Feedbackreglern und den Graphen für das Flüssigkeitsverhalten am Instrument und in der Vorderkammer).

Abbildung 7.4 zeigt den Hauptbildschirm des Maschinentesters. Links oben befindet sich die Fußpedalsteuerung mit vier analogen Achsen und acht digitalen Knöpfen. In der Mitte kann zwischen vitreoretinalen und Katarakt-Trainingsmodul gewählt werden. Über „Feedback“ können die Werte für die Vorderkammer- und Maschinen-Simulation manipuliert werden. Im unteren Bereich werden die Ausgabewerte angezeigt. Im linken Fenster werden die Druckwerte der Maschine angezeigt und im rechten der Druckverlauf in der Test-Chamber. Diese kann unter „Configure“ konfiguriert werden. Hierfür steht eine lineare und eine nichtlineare Variante zur Verfügung. Es lässt sich das Verhältnis von Normalvolumen zu Normaldruck festlegen, sowie die Ober- und Untergrenzen für Volumen und Druck. Die Steigungen für den Druckanstieg bzw. -verlust sind hiermit für die linearen Funktionen genau bestimmt. Für die nichtlineare Compliance-Funktion kann die Steigung der Kurve zusätzlich angepasst werden. In Abb. 7.6 ist dies für eine nichtlineare und in Abbildung 7.5 für eine Lineare Compliance-Funktion dargestellt.

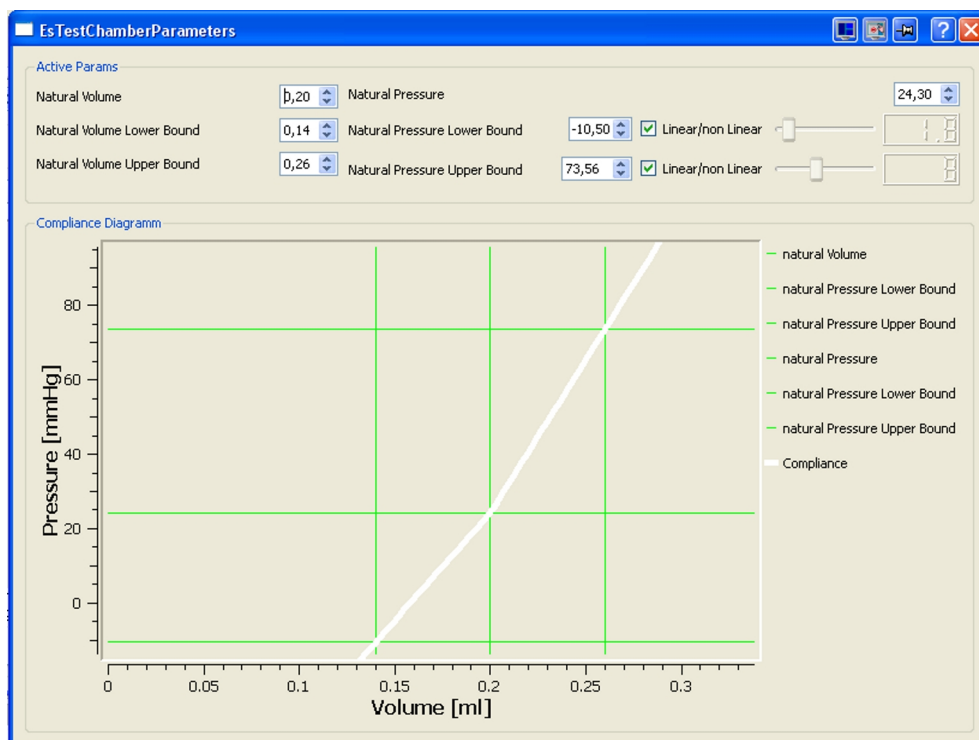


Abbildung 7.5: Der Test-Chamber Konfigurationsdialog mit linearen Einstellungen für den Verlauf der Test-Chamber Compliance-Funktion.

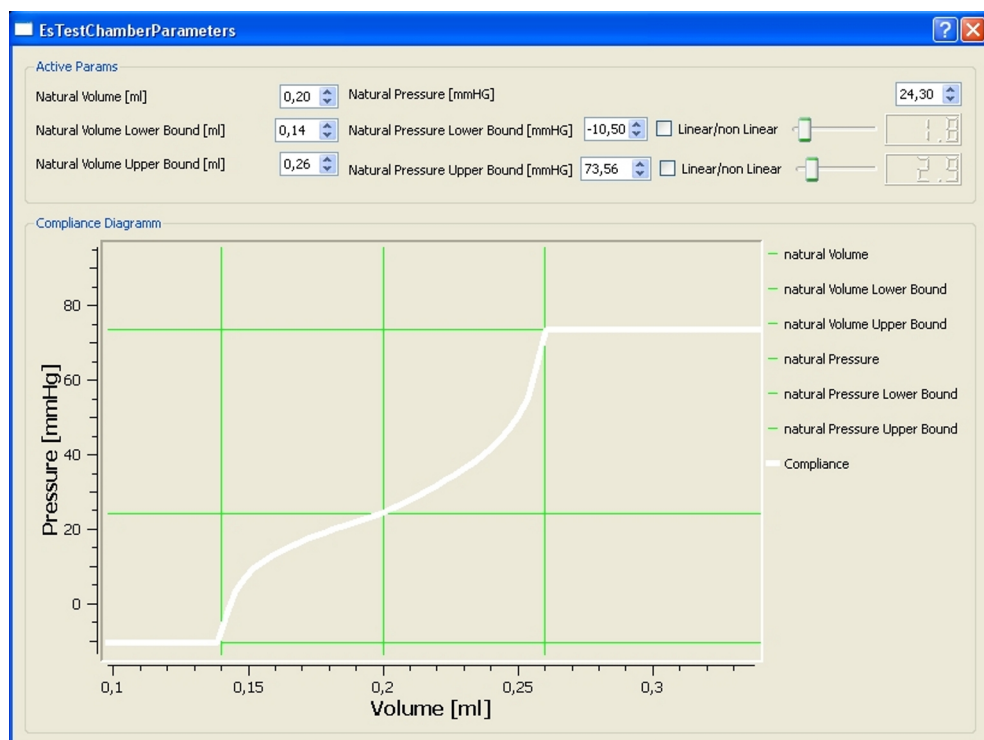


Abbildung 7.6: Der Test-Chamber Konfigurationsdialog mit nichtlinearen Einstellungen für den Verlauf der Test-Chamber Compliance-Funktion.

### 7.4 Diskussion

Ein VR-Simulator kann nicht direkt um reale Komponenten erweitert werden. Diese müssen in die virtuelle Realität übertragen werden. Bei Eyesi sind solche Erweiterungen die Operationsmaschinen, die in der Realität für die Operation benötigt werden. Es wurden geeignete Modelle entwickelt und die Software entsprechend angepasst. Die Erweiterung der Eyesi-Software stellt eine Steigerung der Realitätsnähe des Systems dar. Die Immersion beim Lernenden wird verstärkt und es ist dem Lernenden möglich, sich schon in der virtuellen Umgebung mit der Operationsmaschine vertraut zu machen. Da eine Vielzahl von Operationsmaschinen für die Phakoemulsifikation verfügbar sind, wurde ein Software-Konzept entwickelt, mit dem es möglich ist, verschiedene Operationsmaschinen in die Simulation zu übertragen. Es wurden Schnittstellen zwischen den einzelnen Teilen entwickelt, die die relevanten Werte für den jeweiligen Software-Teil zur Verfügung stellen. So können verschiedene Benutzeroberflächen und verschiedene Maschinenmodelle mit verschiedenen Schlauchsystemen und Instrumentformen kombiniert werden.

Zurzeit basiert die OMS auf einer „Software-Maschine“ und vier verschiedenen Benutzeroberflächen. Maschinenparameter wie die Flaschenständergeschwindigkeit, die Schlauchlänge und der -durchmesser sind in Dateien gespeichert, die beim Programmstart geladen werden. Dies ermöglicht eine Anpassung an verschiedene reale Maschinentypen, obwohl nur eine Implementierung der Maschine benutzt wird.

Durch den modularen Aufbau ist es möglich, einzelne Teile speziell im Quellcode anzupassen und für jede reale Maschine eine eigene Simulation zu erstellen. Dieses Konzept ermöglicht es die Maschinen-Simulation schnell und einfach zu erweitern.



## Trainingsmodule

Eyesi beinhaltet eine Reihe von Trainingsmodulen zur Aus- und Weiterbildung von Augenchirurgen. Diese teilen sich in abstrakte Trainingsmodule und Operationsszenarien [99]. Bei abstrakten Trainingsmodulen wird eine spezielle Aufgabe auf einer abstrakten Ebene präsentiert. Diese gilt es möglichst fehlerfrei zu lösen. Im Gegensatz dazu handelt es sich bei Operationsszenarien um detailreichere Simulationen. Das Operationsgebiet wird so realistisch wie möglich simuliert. Ziel ist es, die gesamte Operation zu erlernen und dabei abzuwägen, welches der richtige Schritt ist.

Derzeit befinden sich zwei Trainingsmodule, die sich mit der Phakoemulsifikation beschäftigen, in der Eyesi-Software. Ein dritter Ansatz wurde nicht weiter entwickelt. Er soll hier dennoch kurz vorgestellt werden.

Das in *Kapitel 6 - Gewebeinteraktion mit dem Instrumentarium* vorgestellte **Druck- und Volumenmodell** für den vorderen Augenabschnitt gehört zu den Basiseigenschaften jedes Trainingsmoduls. Die dort berechneten Werte fließen sowohl in die Visualisierung als auch in die Bewertung ein. Beispiele sind der Kammerunterdruck, der Kammerkollaps und die damit einhergehende Abflachung der Cornea und die Bewegungen der Iris.

### 8.1 Phako - Der erste Prototyp

Die erste Version eines Trainingsmoduls zur Phakoemulsifikation entstand 2004 und wurde zu Demonstrationszwecken präsentiert. Zu Beginn der Entwicklung waren viele Dinge, wie z. B. die Auswirkungen des Ultraschalls, das Druck- und Volumenverhalten und der Einfluss der Maschine auf die Simulation, unklar. Es wurde mit der Entwicklung eines Prototypen begonnen. Die dabei gewonnen Erkenntnisse flossen in die weitere Entwicklung ein. Aus diesem Grund soll dieser Prototyp hier beschrieben werden.

### 8.1.1 Anforderungen

Die Anforderung war die Entwicklung einer prototypischen Phakoemulsifikation als Diskussionsgrundlage und zur Erkenntnisgewinnung. Mit dem Trainingsmodul sollte es möglich sein:

- einen Graben mit Ultraschall zu ziehen,
- die Linse zu rotieren,
- die Linse zu spalten und
- die Linsenstücke vollständig abzusaugen.

### 8.1.2 Umsetzung

Die Linse wurde als Festkörper modelliert, der in eine starre Kapsel eingebettet war. Dabei bestand das Linsenmodell nur aus Oberflächen. Die Linse konnte mit dem Phakoinstrument in der Kapsel rotiert werden. Die Kollisionserkennung und -auflösung erfolgte über eine Raumaufteilung (SSD, siehe 3.4.1.1) und über einen zwangsbasierten Ansatz.

**Ultraschall** Das Grabenziehen mittels Ultraschall wurde durch ein Verschieben der Oberflächendreiecke in z-Richtung erreicht. Unterschritten die beiden Oberflächen einen Mindestabstand wurde die Ober- und Unterseite des Modells miteinander verbunden und es entstand ein Loch.

**Zerteilen** Für das Zerteilen der Linse wurde eine Vorzugsrichtung entlang des Grabens ermittelt. Unter Zuhilfenahme eines zweiten Instruments war das Zerteilen möglich. Die Eindringtiefe der Instrumente in die Seitenwände des Grabens diente als Maß für die Spannung. Überschritt diese einen Schwellwert, wurde die Linse zerteilt, indem Ober- mit Unterseite verbunden wurden. So entstanden zwei unabhängige Teilstücke. Für diese wurden die Festkörper-Parameter neu berechnet.

### 8.1.3 Ergebnisse

Der entwickelte Prototyp ist in Abbildung 8.1 dargestellt. Es sind die verschiedenen Phasen der Phakoemulsifikation dargestellt. In Abbildung 8.1(a) sieht man, wie der Graben durch Verschieben der Oberfläche gezogen wurde. Abbildung 8.1(b) zeigt die entlang des Grabens gespaltene Linse und Abbildung 8.1(c) das Absaugen der restlichen Linse. Mit dem Prototyp war

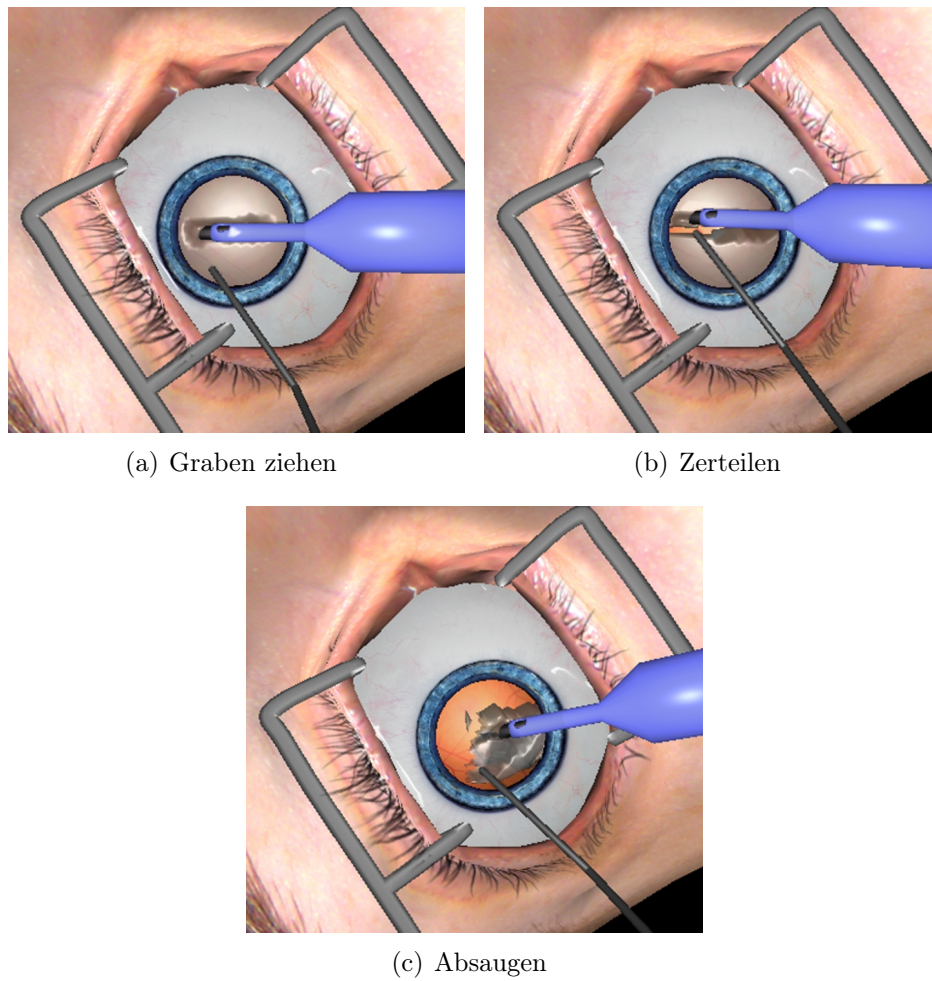


Abbildung 8.1: Screenshots Phako - Der erste Prototyp.

es möglich, eine „Phako - Divide and Conquer“ Operation durchzuführen. Er besaß viele Unzulänglichkeiten. Die Rotation der Linse wurde nur über die Instrumentbewegung berechnet. Dadurch kam es leicht zum Kleben des Linsenmodells am Instrument. Die Rissrichtung wurde nur analytisch anhand der Richtung des Grabens und abhängig von der Instrumentbewegung bestimmt. Dies führte zu spontanen Rissbildungen und unintuitiven Rissrichtungen. Es gab keine Berechnung der Ultraschallenergie. Der Ultraschall wurde aktiviert, sobald das Fußpedal in Stellung **C** war. Das Verschieben der Oberflächendreiecke erzeugte zwar einen visuell ansprechenden Effekt, es konnte damit aber keine Verstopfung und Anhaftung modelliert werden. Es wurden keinerlei Fluss- und Druckberechnungen durchgeführt.

Die Nachteile des Trainingsmoduls waren für eine qualitativ hochwertige Ausbildung nicht tragbar. Deshalb wurden die Entwicklungen in diese Richtung eingestellt.

Die daraus gewonnenen Erkenntnisse waren, dass neben einer plausiblen Interaktion mit den Instrumenten vor allem das Druck- und Strömungsverhalten, das durch die Maschine erzeugt wird, ausschlaggebend für den Lernerfolg ist. Dies führte zu der Entwicklung der Maschinen- und Vorderkammersimulation.

## 8.2 Phako-Training

Beim Phako-Training handelt es sich um ein abstraktes Trainingsmodul, das den Umgang mit der Operationsmaschine und dem Fußpedal vermitteln soll. Operationen am vorderen Augenabschnitt erfordern einen sicheren Umgang in der Benutzung der Operationsmaschine. Dazu gehört das Wissen über die Auswirkungen der einzelnen Maschinenparameter wie Fluss und Vakuum auf das Verhalten in der Vorderkammer. Des Weiteren muss man wissen, welche Folgen eine Fehlbedienung nach sich ziehen kann. Der sichere Umgang mit dem Fußpedal gehört ebenfalls zu den Grundlagen. Das Ziel ist, dass der Lernende erkennt, welche Auswirkungen die Maschinenkonfiguration und die Fußpedalstellung haben. Derzeit gibt es keine vergleichbare Anwendung außerhalb von Eyesi. Andere Projekte beschäftigen sich nur mit der Simulation der kompletten Phakoemulsifikation.

### 8.2.1 Anforderungen

Die Anforderungen ergeben sich aus den oben beschriebenen Eigenschaften, die mit dem Phako-Training gelehrt werden soll.

Das Trainingsmodul erfordert die:

- Modellierung abstrakter Objekte, die Linsenteilstücke darstellen sollen,
- Wahl einer geeigneten Simulation für die Linsenstücke,
- Modellierung der Flüssigkeitsinteraktion mit den abstrakten Objekten,
- Modellierung der Interaktion mit dem Instrumentarium,
- Modellierung der Interaktion mit der Vorderkammer,
- Modellierung der Interaktion der abstrakten Objekte untereinander,
- Modellierung der Ansaugung der abstrakten Objekte,
- Modellierung der Verflüssigung und Absaugung der abstrakten Objekte.

### 8.2.2 Umsetzung

Im Folgenden wird die Umsetzung der einzelnen Anforderungen näher erläutert.

**Abstrakte Objekte** Die Wahl der abstrakten Objekte für das Phako-Training fiel auf Kugeln. Dafür gab es zwei Gründe: der Wiedererkennungswert, denn Kugeln werden in Eyesi oft als abstrakte Objekte verwendet und die Modellierung der Ansaugung der Objekte gestaltete sich für rotationssymmetrische Objekte wesentlich einfacher.

**Geeignete Simulation** Für das Phako-Training wurde ein Mehrkörpersystem mit einem expliziten Integrationsverfahren implementiert, wie es in Abschnitt 3.2.1 beschrieben wurde.

**Flüssigkeitsinteraktion** Für die Strömungssimulation wird der Ansatz von Stam (siehe Abschnitt 3.2.3.1) verwendet, der auf drei Dimensionen erweitert wurde. Als Quellen und Senke für die Simulation dienen die Öffnungen am Phakoinstrument. Für die Strömungsgeschwindigkeit werden die durch die Maschinen-Simulation berechneten Werte verwendet. Das berechnete Vektorfeld der Strömungsgeschwindigkeit wird über die Mobilität  $\mu$  an die simulierten Objekte übertragen. Die Echtzeitbedingung des Trainingsmoduls lässt nur eine Flüssigkeitssimulation in einem Quader mit 18 auf 18 auf 8 Sektoren zu. Diese Flüssigkeitssimulation erzeugt ein plausibles Kraftfeld für die Bewegungen der Kugeln innerhalb der Vorderkammer. Aufgrund der geringen Gitterauflösung bildet sich kein auf die Instrumentspitze gerichtetes

Strömungsfeld aus. Um dennoch eine gerichtete Bewegung zu erzeugen, wird an der Instrumentspitze eine Nahfeld-Interaktion überlagert.

**Kollisionsbehandlung** Es müssen Kollisionen zwischen Kugel und Cornea, Kugel und Kapsel, Kugel und Phakoinstrument sowie Kugel und Kugel erkannt werden. Bei allen Objekten handelt es sich um starre Körper, die als Dreiecksgitter modelliert wurden. Die äußeren Strukturen (Cornea und Kapsel) werden nicht simuliert. Für das Phakoinstrument steht kein Force-Feedback zur Verfügung, eine Kollisionskraft wird nicht benötigt. Eine Kollisionsauflösung wird nur für die Kugeln durchgeführt.

Die Kollisionserkennung wird in eine Broad- und Narrow-Phase unterteilt. Die Wahl für die Broad-Phase fiel auf eine Raumaufteilungsmethode (SSD, siehe Abschnitt 3.4.1.1). Da das mechanische Auge in gewissem Maße beweglich ist, wurde keine fixe SSD im Weltkoordinatensystem gewählt, sondern eine, die sich mit dem vorderem Augenabschnitt mitbewegt. Dadurch ergibt sich die Möglichkeit, nicht simulierte fixe Objekte fest in die SSD einzutragen. Des Weiteren werden nur Knoten der Dreiecksgitter in die Zellen der SSD eingetragen. Durch diese Vereinfachung kann es vorkommen, dass Kollisionen, an denen nur Kanten oder Flächen beteiligt sind, unerkannt bleiben. Die dadurch bei der SSD eingesparte Rechenzeit ermöglicht eine höhere Auflösung der Flüssigkeitssimulation.

In der Narrow-Phase werden die Kollisionen zwischen Knoten und Dreiecken ermittelt. Es werden die Zellen weiter untersucht, in denen sich zwei oder mehr Knoten verschiedener Objekte befinden. Als kollidierend erkannt werden dabei alle Knoten, die einen Mindestabstand  $\epsilon$  zu einem Dreieck unterschreiten oder sich auf der gegenüberliegenden Seite eines Dreiecks befinden. Die Richtung des Dreiecks ist durch seine Visualisierungsrichtung vorgegeben.

Die Kollisionsauflösung erfolgt über einen analytischen zwangsbasierter Ansatz, wie er in Abschnitt 3.4.2.2 beschrieben wurde. Neben den Kollisionspunkte werden die Geschwindigkeiten an den Kollisionspunkten benötigt. Die Kugelgeschwindigkeiten ergeben sich direkt aus der Simulation. Für das Instrument wird die aktuelle Objektgeschwindigkeit aus der Positions Differenz zwischen dem aktuellen und dem letzten Frame ermittelt. Die berechneten Kollisionskräfte werden zur Auflösung der Kugelkollisionen verwendet.

**Aktive Instrumenteigenschaften** Das Ansaugen der Kugeln am Instrument geschieht über ein deskriptives Verfahren. Dabei wird ein Objekt als angesaugt bezeichnet, wenn es sich an der Spitze des Instruments befindet und das Aspirationsvakuum hoch genug ist. Das angesaugte Objekt wird dann nicht mehr simuliert, sondern fest mit dem Instrument verbunden und mit

diesem mitbewegt. Ein festgesaugtes Objekt kann mit Ultraschall verflüssigt und abgesaugt werden. Dafür wurde ein schwellwertbasierter Ansatz gewählt. Jedes Objekt besitzt einen Schwellwert, der die minimale Ultraschallstärke angibt, die aufgebracht werden muss, um das Objekt zu verflüssigen. Für den vom Instrument abgegeben Ultraschall gelten die Berechnungen, wie sie in *Kapitel 5 - Physik der Phakoemulsifikation - Die Operationsmaschine* entwickelt wurden. Die Verflüssigung wird durch eine Verkleinerung des Radius visualisiert. Ist das Objekt klein genug, wird es durch das Instrumentlumen abgesaugt und aus der Simulation entfernt.

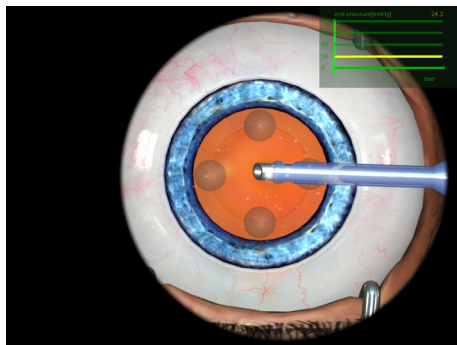
**Repulsion** Die abstoßende Wirkung des Ultraschalls wurde durch ein Lösen der Kugel von der Instrumentspitze modelliert. Ist die Kugel nicht angesaugt, kann sie nicht verflüssigt werden.

**Variationen** Für das Trainingsmodul wurden verschiedene Schwierigkeitsstufen entwickelt. Dies lässt sich über die Objekthärte und die benötigte Ansaugkraft einstellen. Je härter das Objekt ist, desto mehr Ultraschall wird benötigt, um es zu verflüssigen. Zusätzlich muss die Saugstärke vom Lernenden angepasst werden. Wird die Saugstärke zu groß gewählt, kommt es beim Aufbrechen einer Okklusion zum Surge. Der Lernende muss das richtige Verhältnis der Einstellungen zueinander wählen, um die Aufgabe zufriedenstellend zu lösen.

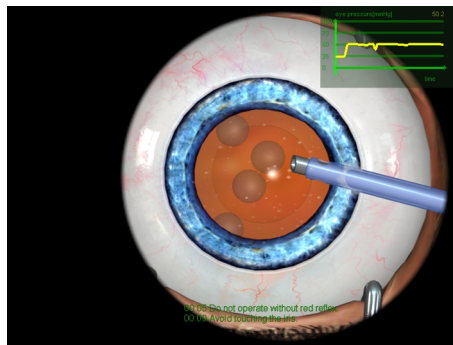
**Scoring** Wurden alle Objekte entfernt, ist das Training beendet und wird ausgewertet. In die Auswertung fließen die Dauer, die verwendete Ultraschallmenge, die ohne Instrumentverstopfung abgegebene Ultraschallmenge, die oben erwähnten Werte der Druckberechnungen und die Berührung empfindlicher Bereiche wie der Cornea oder Iris ein.

### 8.2.3 Ergebnisse

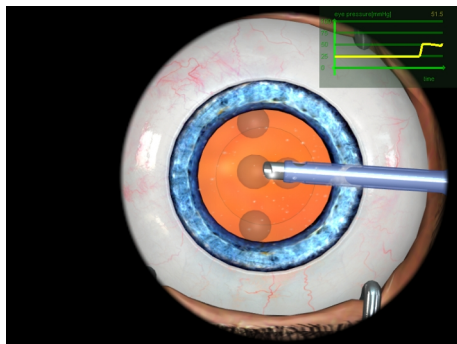
Das entwickelte Trainingsmodul ist in Abbildung 8.2 dargestellt. In Bild 8.2(a) sieht man die Ausgangssituation mit dem eingeführtem Phakoinstrument und der Druckanzeige (oben rechts). Im nächsten Bild 8.2(b) sind die Kugeln durch Aktivieren der Irrigation in Bewegung versetzt worden. In Bild 8.2(c) sieht man, wie eine Kugel festgesaugt wurde und das Handstück verstopft, der Augendruck steigt entsprechend an. Bild 8.2(d) zeigt das Verflüssigen. Dies wird durch eine Verringerung des Radius der Kugel visualisiert. In 8.2(e) sieht man einen Kammerkollaps nach dem Aufbrechen einer Verstopfung. Der rapide Druckabfall ist in der Druckanzeige deutlich zu erkennen. Im letzten Bild 8.2(f) ist das simulierte Flussfeld zu sehen, das für die Bewegung der



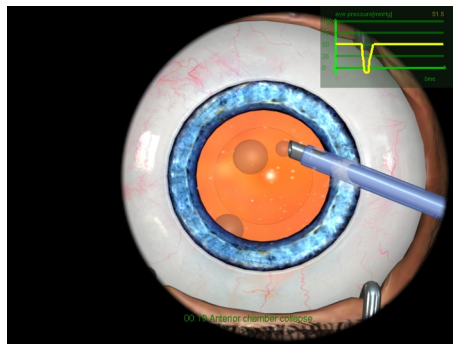
(a) Die Ausgangskonfiguration



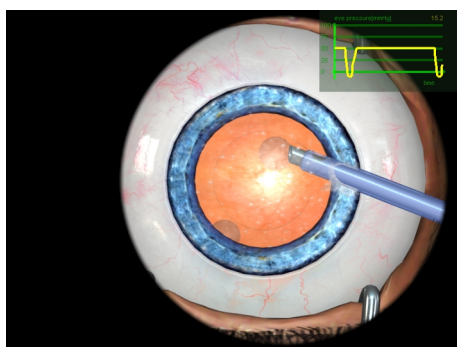
(b) Die Kugeln in Bewegung



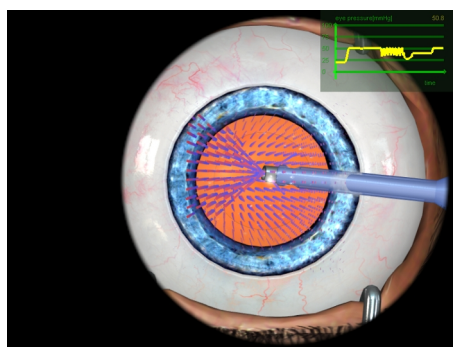
(c) Das Festsaugen



(d) Das Absaugen



(e) Ein Kammerkollaps



(f) Das Flussfeld

Abbildung 8.2: Screenshots des Phako-Trainings.



Kugeln verantwortlich ist. An der Spitze des Instruments ist der Saugkegel deutlich zu erkennen, der der Simulation überlagert wurde.

Das Trainingsmodul wird mit Eyesi ausgeliefert und zur Ausbildung eingesetzt. Als durchweg positiv wurde der hohe Abstraktionsgrad bezeichnet, der es dem Lernenden ermöglicht, sich auf den Umgang mit dem Fußpedal und der Maschine zu konzentrieren.

## 8.3 Phako - Divide and Conquer

Beim Trainingsmodul Phako - Divide and Conquer handelt es sich um eine Operationssimulation, bei der eine spezielle Operationsmethode erlernt werden soll. Das Trainingsmodul wurde vom Autor bereits in [42] und [41] veröffentlicht.

Bei Phako - Divide and Conquer wird zuerst ein Graben mittels Ultraschall durch die Linse gezogen. Mit dem Phakoinstrument und einem weiteren Instrument zerbricht der Chirurg die Linse entlang dieses Grabens. Danach rotiert er die Linse um 90° und wiederholt die Prozedur. Die so entstandenen vier Linsenstücke werden einzeln verflüssigt und abgesaugt. Der Chirurg saugt sie mit dem Phakoinstrument an und bringt sie in die Mitte der Vorderkammer, dort können sie gefahrloser mit Ultraschall verflüssigt und abgesaugt werden. Die Prozedur lässt sich in mehrere Schritte einteilen. Dazu gehören das „Sculpting“ und das „Quadrant Removal“. Das „Sculpting“ bezeichnet den Prozess des Grabenziehens und das „Quadrant Removal“ das Verflüssigen und Absaugen der einzelnen Linsenstücke. Für die einzelnen Schritte ist es notwendig, verschiedene Maschineneinstellungen zu benutzen, um den bestmöglichen Operationserfolg zu erlangen.

Mit der kompletten Simulation der Phakoemulsifikation beschäftigen sich derzeit einige Gruppen. Diese wurden bereits im *Kapitel 4 - Augenchirurgische Simulatoren* erwähnt. Als die Arbeiten zu diesem Trainingsmodul begannen, existierten nur die Arbeiten von Söderbergh et al. [81] und [82].

### 8.3.1 Anforderungen

Die Anforderungen ergeben sich aus der realen Operationsmethode. Im Speziellen erfordert es die:

- Modellierung einer realistisch aussehender Linse,
- Wahl einer geeigneten Gewebesimulation,

- Modellierung einer realistischen Gewebeinteraktion mit dem Instrumentarium,
- Modellierung der Möglichkeit einen Graben durch die Linse zu ziehen,
- Modellierung der Flüssigkeitsinteraktion mit den Teilstücken,
- Modellierung der Absaugung der einzelnen Teilstücken,
- Modellierung der Interaktion mit der Vorderkammer.

### 8.3.2 Umsetzung

Für die Umsetzung wurden die folgenden Modelle und Methoden gewählt.

**Die Linse** Die Modellierung einer realistisch aussehenden Linse hängt mit der Wahl der Gewebesimulation zusammen. Da es sich bei der Linse um ein volumetrisches Objekt handelt, wurde das DEM-Verfahren (siehe Abschnitt 3.2.2) als Simulationsverfahren gewählt. Die Oberflächendreiecke werden dabei zur Visualisierung genutzt. Das DEM-Gitter, das der Linse zugrunde liegt, besitzt einen schalenförmigen Aufbau und spiegelt so ansatzweise den realen Aufbau wider.

**Instrumentinteraktion** Die Modellierung der Gewebeinteraktion mit dem Instrumentarium basieren auf den in *Kapitel 6 - Gewebeinteraktion mit dem Instrumentarium* eingeführten Modellen und wurden entsprechend implementiert.

**Flüssigkeitsinteraktion mit den Teilstücken** Bei der Simulation der Phakoemulsifikation wurde die Interaktion mit der Flüssigkeit über ein deskriptives Modell erreicht. In das Modell fließen die von der Maschinen-Simulation berechneten Irrigations- und Aspirationswerte ein.

Für das Strömungsprofil werden die vektoriellen Funktionen aus *Kapitel 6 - Gewebeinteraktion mit dem Instrumentarium* verwendet, die für jeden Knoten einen Kraftvektor berechnet.

**Ansaugen der Linsenstücke** Für das Ansaugen von Linsenstücken wird das deskriptive Modell aus *Kapitel 6 - Gewebeinteraktion mit dem Instrumentarium* verwendet.

**Absaugen der Linsenstücke** Die herausgelösten Tetraeder werden in das Instrumentinnere gesaugt und aus der Simulation entfernt.

**Simulation der gesamten Vorderkammer** Für die Simulation der Phakoemulsifikation wurde neben der Simulation der Linse eine Simulation der kompletten Vorderkammer entwickelt. Diese beinhaltet Interaktionen mit den einzelnen Gewebeteilen, mit den Instrumenten sowie mit der Linse selbst. So wurde die Kapsel als Feder-Masse-System modelliert, das sich über Instrumentinteraktionen deformieren lässt. Zur Kollisionsbehandlung wurden Kapsel und Cornea als Penalty-Objekte nachgebildet. Diese haben die Form von Kugelschalen und erzeugen bei Durchdringung rückstellende Kräfte für die Linse bzw. die Linsenstücke. Die Kapsel und die dazugehörigen Penalty-Objekte sind über Federn in der Vorderkammer aufgehängt. Kommt es zu tiefen Kollisionen zwischen Instrument und Linse, die zu Instabilitäten der Linsensimulation führen können, ermöglicht eine Verschiebung des gesamten Linsensystems eine Reduzierung dieser Kollision. Besteht zwischen Instrument und Linse keine Kollision mehr, nimmt das Linsensystem wieder seine ursprüngliche Lage ein.

**Variationen** Um einzelne Operationsschritte besser erlernen zu können, wurden vier Variationen entwickelt. Bei der ersten Variante muss ein einzelner Quadrant entfernt werden. Die zweite Variante beinhaltet eine in vier Quadranten aufgespaltene Linse. Es müssen alle Quadranten verflüssigt und abgesaugt werden. In der dritten Variante wird das Zerteilen trainiert. Dabei ist schon ein Graben vorgegeben, entlang diesem muss die Linse mit beiden Instrumenten gespalten werden. Bei der letzten Variante ist die Situation nach der Kapsulorhexis nachgestellt. Es muss eine Phakoemulsifikation der kompletten Linse durchgeführt werden.

**Scoring** In die Auswertung fließen, wie schon beim Phako-Training, allgemeine Ultraschall- und Druckwerte ein. Abhängig von der Ausgangskonfiguration gibt es für das Erfüllen der Aufgabe unterschiedliche Punkteverteilungen. So gibt es für das Absaugen des Gewebes und für das erfolgreiche Zerteilen Punkte.

### 8.3.3 Ergebnisse

Das entwickelte Trainingsmodul ist in Abbildung 8.3 dargestellt. Abbildung 8.3(a) zeigt das Graben ziehen, Abbildung 8.3(b) das Zerteilen der Linse und Abbildung 8.3(c) das Absaugen eines Teilstückes.

Das Trainingsmodul wurde von Medizinern als realistisch eingestuft. Es traten jedoch Probleme bei Selbstkollisionen und dem Steifigkeitsverhalten der Linse auf. Die Probleme können teilweise auf die durch die Echtzeitbedingung

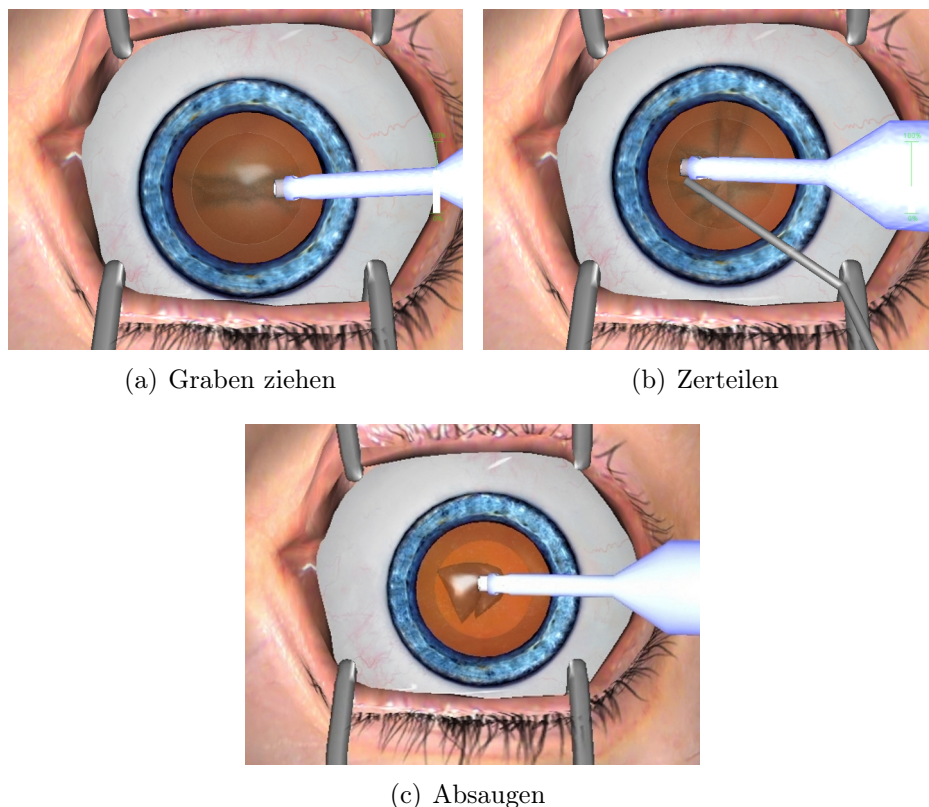


Abbildung 8.3: Screenshots Phako - Divide and Conquer.

begrenzte Rechenzeit zurückgeführt werden. So lässt die begrenzte Rechenzeit eine Überprüfung auf Selbstkollisionen nur für Hüllkörper der entstanden Linsenstücke zu. Auswirkungen sind eine unnatürlich wirkende Kollisionsantwort und Durchdringungen. Diese Probleme könnten durch schnellere Hardware gelöst werden. Auch wurde teilweise die Linse als zu weich angesehen. Die Steifigkeit der Linse könnte durch mehrere Maßnahmen verbessert werden. Eine Erhöhung der Steifigkeitskonstanten wäre eine Möglichkeit, dies erfordert eine Verkleinerung des Zeitschrittes und führt zu einer Erhöhung der Rechenschritte. Dies lässt die begrenzte Rechenzeit zurzeit nicht zu. Es könnte ein anderes volumetrisches Simulationsmodell eingesetzt werden, dies erfordert Anpassungen jeglicher Gewebeinteraktionen. Arbeiten an diesen Problemen könnten das Trainingsmodul in Zukunft verbessern.

Das entwickelte Modul war Teil der Eyesi Version 2.3 und wurde zur ophthalmologischen Ausbildung eingesetzt.

## 8.4 Vitreoretinale Trainingsmodule

Im Rahmen dieser Arbeit wurden keine eigenständigen vitreoretinalen Trainingsmodule entwickelt. Es werden allerdings die in der Maschinen-Simulation (siehe *Kapitel 5 - Physik der Phakoemulsifikation - Die Operationsmaschine*) berechneten Werte für die Instrumentinteraktionen verwendet. Ebenfalls kommt das in *Kapitel 6 - Gewebeinteraktion mit dem Instrumentarium* vorgestellte **Druck- und Volumenmodell** in den vitreoretinalen Trainingsmodule zum Einsatz. Abbildung 8.4(a) zeigt ein vitreoretinales Trainingsmodul mit Normaldruck. Der Chirurg erhält über zwei Merkmale Informationen zum aktuellen Druck im Auge: das **Pulsieren** der Sehnervenscheibe und die **Unschärfe** der Szene. Das Pulsieren entsteht, wenn der Augendruck den Blutdruck überschreitet. Es beginnt beim diastolischen (unteren) Wert und endet mit einer vollkommenen Weißfärbung (siehe Abbildung 8.4(b)) der Adern beim systolischen (oberen) Wert. Bei einem gesunden Menschen liegt der diastolische arterielle Druck zwischen 60 und 85 mmHg (8,0 und 11,3 kPa) und der systolische arterielle Druck zwischen 100 und 130 mmHg (13,3 und 17,4 kPa). Wenn sich der Patient in Ruhelage und flach auf dem Operationstisch befindet, sinken diese Werte entsprechend ab. Nach Rücksprache mit Ärzten wurden für den simulierten Blutdruck folgende Werte gewählt: Für den diastolischen Druck 55 mmHg und für den systolischen Druck 78 mmHg mit einer Pulsdauer von einer Sekunde.

Die Unschärfe entsteht durch ein Abflachen des Auges und der damit einhergehenden Verschiebung der Linse. Dies führt zu einer Defokussierung, wie sie in Abbildung 8.4(c) dargestellt ist.

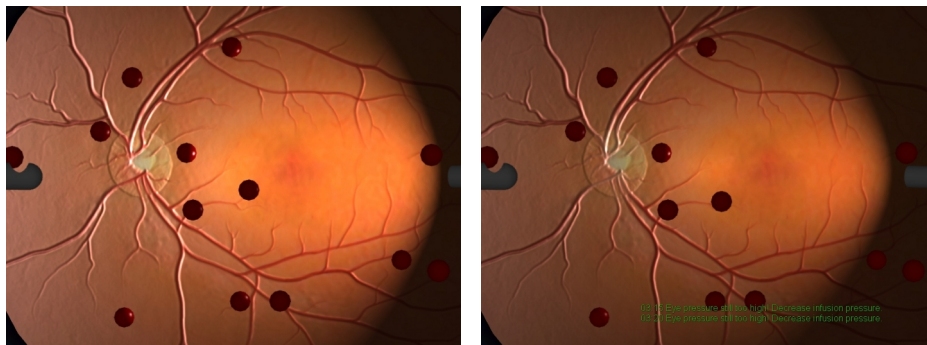
**Abfrage der Werte** Die von der Maschine erzeugten aktiven Instrumenteigenschaften können vom Trainingmodulprogrammierer abgefragt und verwendet werden. Damit lassen sich zum Beispiel die abstrakten Objekte bewegen.

**Setzen der Werte** Weiterhin besteht die Möglichkeit, zusätzliche Werte für die Druck- und Volumenberechnung zu setzen. Dies sind die **Indentation** (Eindellung) und ein externes Volumen durch **Injektion**.

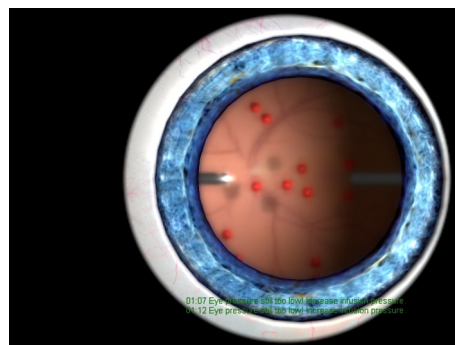
Die Ergebnisse wurden in *Kapitel 6 - Gewebeinteraktion mit dem Instrumentarium* präsentiert.

## 8.5 Diskussion

Die Maschinen- und Hydrodynamiksimulation stellen Grundlagen für jedes Trainingsmodul in Eyesi dar. Dabei besteht eine Trennung zwischen den globalen Druck- und Volumenverhältnissen im Auge (bzw. Vorderkammer)



(b) Auge mit erhöhtem Druck (Weißfärbung der Adern der Sehnervenscheibe)



(c) Auge mit zu geringem Druck

Abbildung 8.4: Screenshots eines vitreoretinalen Trainingsmoduls.

und den Strömungsberechnungen, die durch den Fluss der Flüssigkeit erzeugt werden. Durch eine weitere Verfeinerung der Druck- und Volumensimulation könnte dies in Zukunft vermieden werden.

Die für Eyesi entwickelten Module wurden, beziehungsweise werden, mit Eyesi vertrieben. Es bedarf weiterhin viel Entwicklungsarbeit, um alle Aspekte der Phakoemulsifikation abzudecken.

Das Phako-Training bietet dem Lernenden die Möglichkeit, sich genauer mit den Maschineneinstellungen und deren Auswirkungen in der Vorderkammer zu beschäftigen.

Mit der Simulation der Phakoemulsifikation steht ein Trainingsmodul zur Verfügung, mit dem es möglich ist, die einzelnen Schritte einer Divide and Conquer Kataraktoperation zu erlernen. Schwächen der aktuellen Implementierung sind die grobe Auflösung, sowie die Festlegung auf die spezielle Operationsvariante. Wegen der groben Auflösung ist derzeit keine ausreichend feine Modellierung der einzelnen Schichten, wie Kern und Rinde, möglich.

Geplant ist die Unterstützung anderer Operationsmethoden wie Vertical oder Horizontal Chop. Die daraus gewonnenen Erkenntnisse ermöglichen es vielleicht, die Simulation so weit zu verallgemeinern, dass zukünftige Simulationen nicht mehr auf eine spezielle Operationsmethode festgelegt sind und die deskriptive Modellierung durch eine physikalische abgelöst wird.

Ein weiteres Trainingsmodul wäre ein Irrigations- und Aspirationsmodul, bei dem das Entfernen des Cortex trainiert werden kann. Neben einem neuen virtuellen Instrument muss der Cortex modelliert werden. Für das Instrument muss ebenfalls ein Strömungsprofil entwickelt und für den Cortex eine geeignete Simulation gefunden werden.





## Diskussion und Ausblick

Ziel dieser Arbeit war die Simulation der Phakoemulsifikation im Augenoperationssimulator Eyesi. Während der Entwicklung hat sich herausgestellt, dass die Verwendung der Phakomaschine einen entscheidenden Einfluss auf den Verlauf der Operation hat. Deshalb wurde die Operationsmaschinen-Simulation entwickelt, die ausgehend von realen Maschinen das Verhalten in der virtuellen Realität nachbildet. Im Speziellen wurden Modelle für die Irrigation, die Aspiration und den Ultraschall entwickelt.

Das Modell für die Irrigation liefert einen Druckwert an der Instrumentspitze. Die Auswirkungen dieses Druckwertes werden in der virtuellen Umgebung ausgewertet. Für die Aspiration wurden Modelle für die zwei gängigen Pumpentypen berücksichtigt. Die Modelle für die Pumpen beinhalten die Auswirkungen der rise time und des Ventings. Damit ein einheitliches Modell für das Schlauchsystem entwickelt werden konnte, wird die Vakuum-Pumpe in eine Fluss-Pumpe umgerechnet. Das Schlauchmodell stellt über die Compliance einen Zusammenhang zwischen Schlauchvolumen und -druck her. Damit ist es möglich den Surge, der beim Aufbrechen einer Okklusion entstehen kann, zu berechnen. Nach der Berechnung aller Teilkomponenten liegen die Werte an der Instrumentspitze, die in der virtuellen Realität benötigt werden, vor.

Der von der Maschine erzeugte Ultraschall wird in einem Aktiv- und Inaktivzeit basierten Modell berechnet. Dieses liefert eine vom Instrument pro Frame abgegebene Ultraschallenergie, die in der virtuellen Realität weiterverwendet wird. Es wurden die Standard-Ultraschallmodulationen implementiert. Das Modell erlaubt es, über die Aktiv- und Inaktivzeit weitere maschinenabhängige Modulationen hinzuzufügen.

Die Schnittrate des Vitrektors wirkt sich auf das Flussverhalten an der Instrumentspitze aus. Dies wird im entwickelten Modell direkt in der Maschine berücksichtigt. Der Trainingsmodul-Programmierer kann auf diese Werte zurückgreifen und muss sie nicht mehr in jedem neuen Trainingsmodul selbst berechnen.

In der Operationsmaschinen-Simulation werden alle Schritte sequentiell abgearbeitet und die relevanten Werte der virtuellen Realität und der Maschinen-GUI zur Verfügung gestellt. Dabei wird zwischen Trainingsmodulen für den vorderen und hinteren Bereich unterschieden. Der Simulationsschritt unterscheidet sich dabei in der Berücksichtigung der instrumentspezifischen Eigenschaft und dem Einfluss des Schlauchsystems.

Durch das Einbringen des Instruments in die Vorderkammer bzw. in das Auge entsteht ein gekoppeltes System. Dieses wirkt sich direkt auf den Irrigationsdruck und indirekt auf den Irrigationsfluss an der Instrumentspitze aus. Um dieses Verhalten zu modellieren, wurde ein Compliance basiertes Modell entwickelt. Die Berechnungen für das Verhalten des vorderen und hinteren Abschnitts beruhen auf diesem Modell. Über das Modell lässt sich das Verhältnis von Normalvolumen zu Normaldruck einstellen. Das Verhalten für erhöhtes und erniedrigtes Volumen lässt sich getrennt einstellen. Es wurde eine lineare und eine nichtlineare Compliance-Funktion entwickelt und untersucht. Die lineare Variante liefert Ergebnisse, die sich besser mit den Vergleichswerten aus der Theorie decken. Aus diesem Grund wird sie in Eyesi eingesetzt. Das Modell des hinteren Abschnitts wurde um das Verhalten während einer Indentation erweitert. Abhängig vom Trainingsmodultyp werden die Druck- und Volumenverhältnisse entweder für den vorderen oder hinteren Abschnitt berechnet. Dadurch ist es nicht möglich Komplikationen wie einen Kapselriss mit Linsenverlust zu simulieren. Für eine realistischere Simulation sollte das Modell entsprechend erweitert werden.

Jegliche Gewebeinteraktionen wurden unter dem Gesichtspunkt der Echtzeitbedingung entwickelt. Die daraus entstandenen Modelle berücksichtigen viele Vereinfachung und Spezialfälle. So wurde ein spezieller Ansatz zur Kollisionsbehandlung zwischen der grob aufgelösten virtuellen Linse und Penalty-Objekten entwickelt. Dabei werden die Kollisionskräfte für kollidierende Kanten, Dreiecke und Tetraeder nur für deren Mittelpunkte berechnet und entsprechend aufgeteilt. Es wurde ein deskriptives und eine physikalisch basiertes Modell für die Bewegungen der virtuellen Linse in der flüssigkeitsgefüllten Vorderkammer erstellt. Die beschränkte Rechenzeit bei Echtzeitsimulationen lässt nicht immer den Einsatz der physikalisch basierten Methode zu. So wird im Phako-Training die physikalisch basierte Methode eingesetzt und im Phako - Divide and Conquer die deskriptive. Die in Zukunft zur Verfügung stehende Erhöhung der Rechenleistung könnte dieses Problem lösen. Der für die Maschinen-Simulation benötigte Verstopfungsgrad an der Instrumentspitze wird nicht genau bestimmt, sondern nur über die Kollision zwischen Kontrollpunkten und Linsengewebe ermittelt. Da es noch unklar ist welcher Effekt für das Verflüssigen der Linse verantwortlich ist, wurde ein Modell entwickelt, das das Verhalten auf makroskopischer Ebene nachbildet. Es ist Schwellwert basiert und berücksichtigt den von der Maschine bereitgestellten Ultraschall, die Gewebeparameter der virtuellen Linse, die Entfernung

---

zur Instrumentspitze und die bereits aufgenommene Ultraschallmenge. Überschreitet ein Tetraeder die für ihn zulässige Ultraschallmenge wird er aus dem Simulationsmesh gelöst. Für die Auswirkungen des Vakuums auf das Gewebe wurde ein ähnliches Modell entwickelt. Es besitzt jedoch einen von der bereits aufgenommenen Ultraschallmenge abhängigen Schwellwert. So löst sich durch Ultraschall geschwächtes Gewebe leichter durch Vakuum als nicht geschwächtes. Das für das Absaugen entwickelte Modell wirkt sich auf die aus dem Simulationsmesh gelösten Tetraeder aus. Diese werden ins Instrumentinnere gesaugt und beim Überschreiten einer vorgegebenen Stelle aus der Simulation entfernt. Die verwendeten Modelle decken alle für die Simulation der Phakoemulsifikation benötigten Bereiche ab. Verbesserungen wären eine genauere Kollisionsbehandlung und ein realistischeres Modell für das Ultraschall- und Vakuumverhalten.

Für diese Arbeit wurden drei Trainingsmodule entwickelt: der Phako-Prototyp, das Phako-Training und Phako - Divide and Conquer. Der Phako-Prototyp diente als Produktdemo und wurde auf verschiedenen Messen vorgeführt. Die aufgetretenen Probleme führten zu der Entwicklung der Operationsmaschinen-Simulation und der weiteren Trainingsmodule. Das Phako-Training bietet dem Lernenden die Möglichkeit, sich mit den Auswirkungen der Einstellungen der Maschine und dem Fußpedal vertraut zu machen. Bei der Operationssimulation Phako - Divide and Conquer kann diese Methode der Kataraktoperation erlernt werden. Für jedes Trainingsmodul wurden unterschiedlich schwierige Varianten implementiert. Die Qualität des durchgeführten Trainings wird mit einem Scoring gemessen. Dies ermöglicht eine objektive Bewertung des Lernerfolges.

Zukünftige Arbeiten an Eyesi werden die Erweiterung und Verbesserung der Maschinen-Simulation, des Druck- und Volumenverhaltens des vorderen und hinteren Abschnitts und der Trainingsmodule betreffen.

So muss das Maschinenmodell an die neu erscheinenden realen Maschinen angepasst werden. Dazu gehört die Entwicklung neuer Modelle für spezielle Pumpentypen und die Übertragung neuer maschineninternen Regelmechanismen zur Surge-Unterdrückung. Die Übertragung der Maschinen-GUI gehört ebenfalls zu diesen Aufgaben.

Derzeit führt VRmagic mit weiteren Maschinenherstellern Gespräche über die Portierung ihrer Maschinen-GUI und ihrer speziellen Software-Implementierungen für ihre Maschinenregelung.

Eine Verbesserung für das Druck- und Volumenverhalten wäre, die Simulation für den vorderen und hinteren Augenabschnitt zusammenzufassen. Ebenso könnte eine Verfeinerung der Flüssigkeitssimulation die Entwicklung weiterer deskriptiver Modelle für das Strömungsfeld der Instrumente überflüssig machen.

Bei den Trainingsmodulen könnten Arbeiten an den Modulen selbst und die Entwicklung neuer Module die Qualität der Ausbildung weiter steigern. Für die Simulation der Phakoemulsifikation könnten Trainingsmodule für die anderen Operationsmethoden entwickelt werden. Als neues Modul könnte man sich die Entwicklung eines Irrigations- und Aspirationsmoduls zur Rindenentfernung vorstellen. Das Einsetzen der Kunstlinse stellt eine Möglichkeit für ein eigenes Modul dar. Neben Trainingsmodulen für Teilschritte der Operation wäre ein Modul wünschenswert, mit dem die komplette Operation durchgeführt werden kann.

# Abbildungsverzeichnis

2.1	Die Anatomie des Auges . . . . .	8
2.2	Die Linse . . . . .	9
2.3	Die Katarakt . . . . .	10
3.1	Approximationsgüte . . . . .	32
4.1	Melerit PhacoVision . . . . .	43
4.2	EYESIM . . . . .	45
4.3	SOFA-EYE . . . . .	46
4.4	SOFA-Cataract . . . . .	47
4.5	Eyesi . . . . .	48
4.6	Eyesi - Köpfe . . . . .	49
4.7	Eyesi - Fußpedal . . . . .	50
5.1	Phakohandstück . . . . .	56
5.2	Aktuelle Phakomaschinen . . . . .	59
5.3	Phakomaschine - Touchscreen . . . . .	60
5.4	Phakomaschine - Fußpedal . . . . .	62
5.5	Schaltpunkte - Einfach-Linear . . . . .	63
5.6	Schaltpunkte - Zweifach-Linear . . . . .	64
5.7	Die Peristaltik-Pumpe . . . . .	65
5.8	Die Venturi-Pumpe . . . . .	66
5.9	Phakohandstück - Schematisch . . . . .	68
5.10	Compliance . . . . .	78
5.11	Instrument Unverstopft . . . . .	79
5.12	Beginn der Verstopfung . . . . .	79
5.13	Vollständige Verstopfung . . . . .	80
5.14	Verstopfung bricht auf . . . . .	81
5.15	Rise time Peristaltik-Pumpe . . . . .	93
5.16	Rise time Venturi-Pumpe . . . . .	94
5.17	Schlauchvolumen mit $k = 50$ mmHg . . . . .	95
5.18	Schlauchvolumen mit $k = 500$ mmHg . . . . .	96
5.19	Schlauchvolumen mit $k = 1000$ mmHg . . . . .	97
5.20	Ultraschallmodulation - Linear . . . . .	98
5.21	Ultraschallmodulation - Linear und Fix . . . . .	99
5.22	Ultraschallmodulation - Pulse Mode . . . . .	99
5.23	Ultraschallmodulation - Burst Mode . . . . .	100

5.24	Ultraschallmodulation - Hyperpulse . . . . .	100
5.25	Verhältnis Flow zu Cutrate . . . . .	101
6.1	Der Gesamtaufbau . . . . .	106
6.2	Missverhältnis Instrumentgröße zu Mesh-Auflösung . . . . .	107
6.3	Das Flussfeld des Phakoinstruments . . . . .	115
6.4	Die Koordinatenachsen des Phakoinstruments . . . . .	116
6.5	Die Interaktionsbereiche des Phakoinstruments . . . . .	117
6.6	Ein Irrigationsbereich des Phakoinstruments . . . . .	117
6.7	Der ferne Aspirationsbereich des Phakoinstruments . . . . .	118
6.8	Der nahe Aspirationsbereich des Phakoinstruments . . . . .	119
6.9	Volumenobergrenze lineare Compliance . . . . .	126
6.10	Volumenobergrenze nichtlineare Compliance . . . . .	128
6.11	Druckverlauf nach Seibel . . . . .	129
6.12	Surge bei linearer Compliance . . . . .	130
6.13	Surge bei nichtlinearer Compliance . . . . .	131
6.14	Surge nach Seibel . . . . .	132
6.15	Surge im Vergleich mit Seibel . . . . .	133
6.16	Druck und Volumen bei Indentation . . . . .	135
7.1	Datenabhängigkeiten . . . . .	140
7.2	Aufbau und Einsatz der <i>EsHydrodynamics</i> Klassen . . . . .	145
7.3	VRmagic Maschinen-GUI . . . . .	146
7.4	VRmagic Maschinentester . . . . .	147
7.5	Test-Chamber Parameter linear . . . . .	148
7.6	Test-Chamber Parameter nichtlinear . . . . .	149
8.1	Screenshots Phako - Der erste Prototyp. . . . .	153
8.2	Screenshots des Phako-Trainings. . . . .	158
8.3	Screenshots Phako - Divide and Conquer. . . . .	162
8.4	Screenshots eines vitreoretinalen Trainingsmoduls. . . . .	164

# Tabellenverzeichnis

5.1	Maschinenwertebereich . . . . .	70
5.2	Reale Schläuche . . . . .	89
5.3	Verhältnis Fluss zu Flaschenhöhe . . . . .	90
5.4	Flusswiderstand nach Seibel . . . . .	91
5.5	Flusswiderstand der OMS . . . . .	91
5.6	Verhältnis Fluss zu Vakuum . . . . .	92





# Literaturverzeichnis

- [1] M. Agus, E. Gobbetti, G. Pintore, G. Zanetti, and A. Zorcolo. Real-time cataract surgery simulation for training. In *Eurographics Italian Chapter Conference*, Conference held in Catania, Italy, 2006. Eurographics Association.
- [2] M. Agus, E. Gobbetti, G. Pintore, G. Zanetti, and A. Zorcolo. Real time simulation of phaco-emulsification for cataract surgery training. In *Workshop in Virtual Reality Interactions and Physical Simulations (VRIPHYS 2006)*. Eurographics Association, November 2006. Conference held in Madrid, Spain, November 6-7.
- [3] R. Ahuja. Cataract in human eye. Wikimedia Commons, Dezember 2005. [http://de.wikipedia.org/w/index.php?title=Datei:Cataract\\_in\\_human\\_eye.png](http://de.wikipedia.org/w/index.php?title=Datei:Cataract_in_human_eye.png).
- [4] G. U. Auffarth. *Kataraktchirurgie*. UNI-MED, 1 edition, 2008.
- [5] S. Bandi and D. Thalmann. An adaptive spatial subdivision of the object space for fast collision detection of animated rigid bodies. *Computer Graphics Forum*, 14(3):259–270, 1995.
- [6] D. Baraff. Analytical methods for dynamic simulation of non-penetrating rigid bodies. In *Computer Graphics*, pages 223–232. ACM, 1989.
- [7] D. Baraff. Fast contact force computation for nonpenetrating rigid bodies, 1994.
- [8] D. Baraff and A. Witkin. Physically based modeling: Principles and practice. In *Online Siggraph '97 Course notes*. ACM Press, 1997.
- [9] D. Baraff and A. Witkin. Large steps in cloth simulation. *Computer Graphics*, 32(Annual Conference Series):43–54, 1998.
- [10] R. Barzel, J. F. Hughes, and D. N. Wood. Plausible motion simulation for computer graphics animation. In *Proc. Eurographics Workshop Computer Animation and Simulation*, pages 183–197. Springer, 1996.
- [11] S. Bauer, A. Lyubimov, and P. Tovstik. On the mathematical simulation of the measuring of the intraocular pressure by maklakov method. *TECHNISCHE MECHANIK*, 24(3-4):231, 2004.

- [12] L. Bond, M. Flake, B. Tucker, K. Judd, and M. Boukhny. Physics of phacoemulsification. In *World Congress on Ultrasonics*, pages 169–172, 2003.
- [13] M. Bro-Nielsen and S. Cotin. Real-time volumetric deformable models for surgery simulation using finite elements and condensation. In *Computer Graphics Forum*, pages 57–66, 1996.
- [14] C. D. Bruyns, S. Senger, A. Menon, K. Montgomery, S. Wildermuth, and R. Boyle. A survey of interactive mesh-cutting techniques and a new method for implementing generalized interactive mesh cutting using virtual tools. *The Journal of Visualization and Computer Animation*, 13(1):21–42, 2002.
- [15] G. Burdea and P. Coiffet. *Virtual Reality Technologie*. John Wiley & Sons, 1993.
- [16] R. Caesar and L. Benjamin. *PHACOEMULSIFICATION STEP BY STEP*. Butterworth-Heinemann, 2003.
- [17] J. D. Cohen, M. C. Lin, D. Manocha, and M. K. Ponamgi. I-collide: An interactive and exact collision detection system for large-scale environments. In *Symposium on Interactive 3D Graphics*, pages 189–196, 218, 1995.
- [18] U. Devgan. *Ophthalmology*, chapter Phaco Fluidics & Phaco Ultrasound Power Modulations, pages 457–68. Clinics of North America, 2006 Dec.
- [19] S. Diederich. Prototypischer Aufbau eines Simulators für Eingriffe im Bauchraum. Master’s thesis, Universität Mannheim, 2006.
- [20] D. H. Eberly. *Game Physics*. Elsevier, 2004.
- [21] N. El-Far, S. Nourian, J. Zhou, A. Hamam, X. Shen, and N. Georganas. A cataract tele-surgery training application in a haptic-visual collaborative environment running over the canarie photonic network. In *HAVE 2005 - IEEE Intl. Workshop on Haptic, Audio and Visual Environments and their Applications*, pages 29–32, October 2005.
- [22] C. Ericson. *Real-Time Collision Detection (The Morgan Kaufmann Series in Interactive 3-D Technology)*. Morgan Kaufmann, December 2004.
- [23] N. Foster and D. Metaxas. Practical animation of liquids. In *Graphical Models and Image Processing*, pages 23–30, 1996.
- [24] A. Fuhrmann, G. Sobottka, and C. Groß. Distance fields for rapid collision detection in physically based modeling. In *International Conference on Computer Graphics and Vision (Graphicon)*, 2003.

- [25] S. F. Gibson. 3d chainmail: a fast algorithm for deforming volumetric objects. In *SI3D '97: Proceedings of the 1997 symposium on Interactive 3D graphics*, pages 149–ff., New York, NY, USA, 1997. ACM.
- [26] S. F. F. Gibson and B. Mirtich. A survey of deformable modeling in computer graphics. Technical report, Mitsubishi Electric Research Laboratories, 1997.
- [27] E. G. Gilbert, D. W. Johnson, and S. S. Keerthi. A fast procedure for computing the distance between complex objects in three-dimensional space. *Robotics and Automation, IEEE Journal of*, 4(2):193–203, 1988.
- [28] R. A. Gingold and J. J. Monaghan. Smoothed particle hydrodynamics - theory and application to non-spherical stars. *Mon. Not. Roy. Astron. Soc.*, 181:375–389, November 1977.
- [29] S. Gottschalk, M. C. Lin, and D. Manocha. Obbtree: A hierarchical structure for rapid interference detection. *Computer Graphics*, 30(Annual Conference Series):171–180, 1996.
- [30] N. K. Govindaraju, S. Redon, M. C. Lin, and D. Manocha. Cullide: interactive collision detection between complex models in large environments using graphics hardware. In *HWWS '03: Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware*, pages 25–32, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.
- [31] A. Greß, M. Guthe, and R. Klein. GPU-based collision detection for deformable parameterized surfaces. *Computer Graphics Forum*, 25(3):497–506, Sept. 2006.
- [32] J. Grimm. *Interaktive Echtzeitmodellierung von biologischem Gewebe für Virtuelle Realitäten in der medizinischen Ausbildung*. PhD thesis, Universität Mannheim, 2005.
- [33] J. Grimm, C. Wagner, and R. Männer. Interactive real-time simulation of the internal limiting membrane. In *International Symposium on Medical Simulation*, pages 153–160, MA, June 2004. Cambridge.
- [34] J. K. Hahn. Realistic animation of rigid bodies. In *SIGGRAPH '88: Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, pages 299–308, New York, NY, USA, 1988. ACM.
- [35] G. D. Hart and B. M. Riviere. A constraint-stabilized time-stepping approach for rigid multibody dynamics with joints, contact and friction. In *International Journal for Numerical Methods in Engineering*, pages 2335–2371, 2002.

- [36] S. Hasegawa, N. Fujii, Y. Koike, and M. Sato. Real-time rigid body simulation based on volumetric penalty method. *Haptic Interfaces for Virtual Environment and Teleoperator Systems, International Symposium on*, 0:326, 2003.
- [37] B. Heidelberger, M. Teschner, R. Keiser, M. Müller, and M. Gross. Consistent penetration depth estimation for deformable collision response. In *Proceedings of Vision, Modeling, Visualization VMV 04*, pages 339–346, 2004.
- [38] G. Hirota, S. Fisher, and M. C. Lin. Simulation of non-penetrating elastic bodies using distance fields.
- [39] P. M. Hubbard. Approximating polyhedra with spheres for time-critical collision detection. *ACM Transactions on Graphics*, 15:179–210, 1996.
- [40] J. Huculak and S. Valencia. Vitrectomy Probe: The Affect of Cut Rate on Aspiration Flow Rate. *Invest. Ophthalmol. Vis. Sci.*, 45(5):2018–, 2004.
- [41] O. Jakubik, J. Grimm, and R. Männer. Simulation der Phakoemulsifikation zur Ausbildung von Ophthalmologie-Chirurgen. In *Biomedizinische Technik*, volume 52, Ergänzungsband. BMT, 2007.
- [42] O. Jakubik, R. Männer, and F. Koch. *Health Academy Ausgabe 13 - Modellgestützte Therapie Modellbasierte Methoden zur Ausbildung von Chirurgen in der Ophthalmologie*, volume 13, pages 167 – 184. W. Niederlag (Dresden), H. U. Lemke (Berlin), J. Meixensberger (Leipzig), M. Baumann (Dresden), 2008.
- [43] D. L. James and D. K. Pai. Artdefo: accurate real time deformable objects. In *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 65–72, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co.
- [44] D. M. Kaufman, T. Edmunds, and D. K. Pai. Fast frictional dynamics for rigid bodies. *ACM Trans. Graph*, 24:946–956, 2005.
- [45] J. B. Keller. Impact with friction. *Journal of Applied Mechanics*, 53:1–4, March 1986.
- [46] J. T. Klosowski, M. Held, J. S. B. Mitchell, H. Sowizral, and K. Zikan. Efficient collision detection using bounding volume hierarchies of k-dops. *IEEE Transactions on Visualization and Computer Graphics*, 4:21–36, 1998.
- [47] T. Larsson and T. Akenine-Möller. Collision detection for continuously deforming bodies. In *Eurographics*, pages 325–333, 2001.

- [48] C.-G. Laurell, L. Nordh, E. Skarman, M. Andersson, and P. Nordqvist. Computer-simulated phacoemulsification. In F. Manns, P. G. Soederberg, and A. Ho, editors, *Ophthalmic Technologies XI*, volume 4245, pages 174–176. SPIE, 2001.
- [49] M. C. Lin. Accelerated geometric queries and physical simulation using graphics processors. In *PG '04: Proceedings of the Computer Graphics and Applications, 12th Pacific Conference*, pages 37–39, 2004.
- [50] M. C. Lin and J. F. Canny. A fast algorithm for incremental distance calculation. In *IEEE International Conference on Robotics and Automation*, pages 1008–1014, 1991.
- [51] M. C. Lin and J. F. Canny. Efficient collision detection for animation. In *Eurographics Workshop on Animation and Simulation*, 1992.
- [52] M. C. Lin and S. Gottschalk. Collision detection between geometric models: A survey. In *Proc. of IMA Conference on Mathematics of Surfaces*, pages 37–56, 1998.
- [53] P. Lötstedt. Mechanical systems of rigid bodies subject to unilateral constraints. *SIAM Journal on Applied Mathematics*, 42(2):281–296, 1982.
- [54] L. B. Lucy. A numerical approach to the testing of the fission hypothesis. *Astron. J.*, 82:1013–1024, December 1977.
- [55] S. Melax. Dynamic plane shifting bsp traversal. In *Graphics Interface*, pages 213–220, May 2000.
- [56] B. Mirtich. Hybrid simulation: Combining constraints and impulses. In *Proceedings of First Workshop on Simulation and Interaction in Virtual Environments*. Press, 1996.
- [57] B. Mirtich. *Impulse-based Dynamic Simulation of Rigid Body Systems*. PhD thesis, University of California, Berkeley, December 1996.
- [58] B. Mirtich. V-clip: fast and robust polyhedral collision detection. *ACM Trans. Graph.*, 17(3):177–208, July 1998.
- [59] B. Mirtich. Timewarp rigid body simulation. In *Proc. of ACM SIGGRAPH*, pages 193–200, 2000.
- [60] B. Mirtich and J. Canny. Impulse-based dynamic simulation, 1995.
- [61] M. Müller, D. Charypar, and M. Gross. Particle-based fluid simulation for interactive applications. In *SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 154–159, Aire-la-Ville, Switzerland, 2003. Eurographics Association.

- [62] M. Müller and M. Gross. Interactive virtual materials. In *Graphics Interface (GI 2004)*, pages 239–246, London, Ontario, Canada, May 17–19 2004.
- [63] M. Müller, B. Heidelberger, M. Teschner, and M. Gross. Meshless deformations based on shape matching. *ACM Trans. Graph.*, 24:471–478, 2005.
- [64] M. Moore and J. Wilhelms. Collision detection and response for computer animation. In *Computer Graphics*, pages 289–298, 1988.
- [65] L. C. Moorhead, T. W. Gardner, H. M. Lambert, R. E. O’Malley, A. W. Willis, L. S. Meharg, and W. D. Moorhead. Dynamic intraocular pressure measurements during vitrectomy. *Arch Ophthalmol*, 123(11):1514–1523, 2005.
- [66] M. Müller, R. Keiser, A. Nealen, M. Pauly, M. Gross, and M. Alexa. Point based animation of elastic, plastic and melting objects. In *SCA ’04: Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 141–151, Aire-la-Ville, Switzerland, Switzerland, 2004. Eurographics Association.
- [67] A. Nealen, M. Müller, R. Keiser, E. Boxerman, and M. Carlson. Physically based deformable models in computer graphics. *Computer Graphics Forum*, 25(4):809–836, December 2006.
- [68] S. Nourian, X. Shen, and N. Georganas. Xpheve: An extensible physics engine for virtual environments. In *IEEE Canadian Conference on Electrical and Computer Engineering*, Ottawa, ON, Canada, May 2006.
- [69] J. F. O’Brien and J. K. Hodgins. Dynamic simulation of splashing fluids, 1995.
- [70] M. H. Overmars. Point location in fat subdivisions. *Inf. Process. Lett.*, 44(5):261–265, 1992.
- [71] R. B. Packard. New phaco technology: Is now the time to upgrade?, 2008.
- [72] S. M. Platt and N. I. Badler. Animating facial expressions. *SIGGRAPH Comput. Graph.*, 15(3):245–252, 1981.
- [73] S. Redon, A. Kheddar, and S. Coquillart. Fast continuous collision detection between rigid bodies. In *Proc. of Eurographics (Computer Graphics Forum)*, page 2002, 2002.
- [74] W. T. Reeves. Particle systems - a technique for modeling a class of fuzzy objects. *ACM Transactions on Graphics*, 2:359–376, 1983.
- [75] E. G. Robert. Nonconvex rigid bodies with stacking, 2003.

- [76] M. Schill. *Biomechanical Soft Tissue Modeling - Techniques, Implementation and Applications*. PhD thesis, Universität Mannheim, 2001.
- [77] M. A. Schill, S. F. F. Gibson, H.-J. Bender, and R. Männer. Biomechanical simulation of the vitreous humor in the eye using an enhanced chainmail algorithm. In *MICCAI '98: Proceedings of the First International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 679–687, London, UK, 1998. Springer-Verlag.
- [78] M. Schünke, E. Schulte, and U. Schumacher. *Prometheus: LernAtlas der Anatomie - Kopf und Neuroanatomie*. Thieme, Stuttgart, 2005. Unter Mitarb. von Markus Voll ; Karl Wesker.
- [79] S. Scholtz. Kataraktchirurgie und Kunstlinsen. *DOZ*, 10:64–68, 2005.
- [80] B. S. Seibel. *Phacodynamics: Mastering the Tools and Techniques of Phacoemulsification Surgery, Fourth Edition*. Slack Incorporated, 2004.
- [81] P. G. Söderberg, C.-G. Laurell, D. Artzen, L. Nordh, E. Skarma, P. Nordqvist, and M. Andersson. Computer-simulated phacoemulsification improvements. In F. Manns, P. G. Soederberg, and A. Ho, editors, *Ophthalmic Technologies XII*, volume 4611, pages 76–80. SPIE, 2002.
- [82] P. G. Söderberg, C.-G. Laurell, P. Nordqvist, E. Skarman, and L. Nordh. Virtual cataract surgery: clinical evaluation. In F. Manns, P. G. Sýderberg, and A. Ho, editors, *Ophthalmic Technologies XIII*, volume 4951, pages 62–66. SPIE, 2003.
- [83] P. G. Söderberg, C.-G. Laurell, W. Simawi, P. Nordqvist, E. Skarman, and L. Nordh. Virtual reality phacoemulsification: a comparison between skilled surgeons and students naive to cataract surgery. In F. Manns, P. G. Soederberg, A. Ho, B. E. Stuck, and M. Belkin, editors, *Ophthalmic Technologies XV*, volume 5688, pages 164–169. SPIE, 2005.
- [84] P. G. Söderberg, C.-G. Laurell, W. Simawi, P. Nordqvist, E. Skarman, and L. Nordh. Evaluation of response variables in computer-simulated virtual cataract surgery. In F. Manns, P. G. Söderberg, and A. Ho, editors, *Ophthalmic Technologies XVI*, volume 6138, page 61381B. SPIE, 2006.
- [85] P. G. Söderberg, C.-G. Laurell, W. Simawi, E. Skarman, L. Nordh, and P. Nordqvist. Measuring performance in virtual reality phacoemulsification surgery. In F. Manns, P. G. Soderberg, A. Ho, B. E. Stuck, and M. Belkin, editors, *Ophthalmic Technologies XVIII*, volume 6844, page 684412. SPIE, 2008.

- [86] P. G. Söderberg, C.-G. Laurell, W. Simawi, E. Skarman, P. Nordqvist, and L. Nordh. Performance index for virtual reality phacoemulsification surgery. In F. Manns, P. G. Soederberg, A. Ho, B. E. Stuck, and M. Belkin, editors, *Ophthalmic Technologies XVII*, volume 6426, page 64261B. SPIE, 2007.
- [87] J. Spillmann and M. Teschner. Contact surface computation for coarsely sampled deformable objects. In *Vision, Modeling, Visualization VMV'05*, pages 289–296, Erlangen, Germany, November 2005.
- [88] J. Stam. Stable fluids. In *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 121–128, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co.
- [89] D. Steinemann. *Interactive simulation of topological changes on deformable objects*. PhD thesis, ETH, Zürich, 2008.
- [90] Talos. Eye scheme.svg. Wikimedia Commons, Januar 2008. [http://de.wikipedia.org/w/index.php?title=Datei:Eye\\_scheme.svg](http://de.wikipedia.org/w/index.php?title=Datei:Eye_scheme.svg).
- [91] D. Terzopoulos, J. Platt, A. Barr, and K. Fleischert. Elastically deformable models. *Computer Graphics*, 21:205–214, 1987.
- [92] M. Teschner, S. Girod, and B. Girod. Direct computation of nonlinear soft-tissue deformation. In *Vision, Modeling, and Visualization*, pages 383–390, 2000.
- [93] M. Teschner, S. Girod, and B. Girod. Realistic modeling of elastomechanical properties of soft tissue and its evaluation, 2001.
- [94] M. Teschner, B. Heidelberger, M. Müller, and M. Gross. A versatile and robust model for geometrically complex deformable solids. In *Proc. Computer Graphics International*, pages 312–319, 2004.
- [95] M. Teschner, B. Heidelberger, M. Müller, D. Pomeranets, and M. Gross. Optimized spatial hashing for collision detection of deformable objects. In *Vision, Modeling, Visualization VMV'03*, pages 47–54, 2003.
- [96] M. Teschner, S. Kimmerle, B. Heidelberger, G. Zachmann, L. Raghupathi, A. Fuhrmann, M.-P. Cani, F. Faure, N. Magnenat-Thalmann, W. Strasser, and P. Volino. Collision detection for deformable objects, 2004.
- [97] D. Tracey. *ANATOMICA*. Könnemann, 2004.
- [98] G. van den Bergen. Efficient collision detection of complex deformable models using aabb trees. *Journal of graphics, gpu, and game tools*, 2(4):1–14, 1997.



- [99] C. Wagner. *Virtuelle Realitäten für die chirurgische Ausbildung: Strukturen, Algorithmen und ihre Anwendung*. PhD thesis, Universität Mannheim, 2003.
- [100] K. Waters. A muscle model for animation three-dimensional facial expression. In *SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 17–24, New York, NY, USA, 1987. ACM.
- [101] M. Wenzel, D. Pham, A. Scharrer, K. Schayan, and J. Klasen. Ambulante Intraokularchirurgie: Ergebnisse der Umfrage 2008 von BDOC, BVA und der DGII Rückblick über die letzten 20 Jahre. *OPHTHALMO CHIRURGIE*, 21:199–211, 2009.
- [102] W. S.-K. Wong and G. Baciú. Hardware-based collision and self-collision for rigid and deformable surfaces. *Presence: Teleoper. Virtual Environ.*, 13(6):681–691, 2004.
- [103] wre. Kataraktoperation: Risikominderung einer Blutung bei oraler Antikoagulation. *Dtsch Arztebl*, 102(1-2):A–58–, 2005.



# Abkürzungsverzeichnis

AABB	engl. <b>A</b> xis- <b>A</b> ligned <b>B</b> ounding <b>B</b> ox
ABS	engl. <i><b>A</b>spiration <b>B</b>ypass <b>S</b>ystem</i>
ALP	engl. <i><b>A</b>spiration <b>L</b>ine <b>P</b>ressure</i>
AVG	engl. <i><b>a</b>verage phaco power</i>
BEM	engl. <i><b>B</b>oundary <b>E</b>lement <b>M</b>ethod</i>
BIOM	<b>B</b> inokulares <b>I</b> ndirektes <b>O</b> phthalmomikroskop
BSP	engl. <i><b>B</b>inary <b>S</b>pace <b>P</b>artitioning</i>
BSS	engl. <i><b>B</b>alanced <b>S</b>alt <b>S</b>olution</i>
BVH	engl. <b>B</b> ounding <b>V</b> olume <b>H</b> ierarchy
CASE	engl. <i><b>ChA</b>MBER <b>S</b>tabilization <b>E</b>nvironment</i>
CFR	engl. <i><b>C</b>utrate <b>F</b>low <b>R</b>eduction</i>
cmH <sub>2</sub> O	Einheit für den Druck in Höhe Zentimeter-Wassersäule
CRS4	<b>C</b> enter for <b>A</b> dvanced <b>S</b> tudies, <b>R</b> esearch and <b>D</b> evelopment in <b>S</b> ardinia
CUDA	engl. <b>C</b> ompute <b>U</b> nified <b>D</b> evice <b>A</b> rchitecture
DEM	engl. <b>D</b> eformation <b>E</b> nergy <b>M</b> inimization
DISCOVER	engl. The <b>D</b> istributed & <b>C</b> ollaborative <b>V</b> irtual <b>E</b> nviron- ments <b>R</b> esearch <b>L</b> aboratory
DOG	<b>D</b> eutsche <b>O</b> phthalmologische <b>G</b> esellschaft (DOG)
DOP	engl. <b>D</b> iscrete <b>O</b> rientation <b>P</b> olytope
ECCE	engl. <i><b>E</b>xtra <b>C</b>apsular <b>C</b>ataract <b>E</b>xtraction</i>
EPT	engl. <i><b>e</b>ffective phaco <b>t</b>ime</i>
Eyesi	engl. <i><b>E</b>ye <b>S</b>urgery <b>S</b>imulator</i>
FDM	<b>F</b> inite <b>D</b> ifferenzen <b>M</b> ethode
FEM	<b>F</b> inite <b>E</b> lemente <b>M</b> ethode
FVM	<b>F</b> inite <b>V</b> olumen <b>M</b> ethode
GPGPU	engl. <b>G</b> eneral- <b>P</b> urpose computation on <b>G</b> raphics <b>P</b> roces- sing <b>U</b> nits
GUI	engl. <i><b>G</b>raphical <b>U</b>ser <b>I</b>nterface</i>
HAVE	<b>H</b> apto- <b>A</b> udio- <b>V</b> isual <b>E</b> nvironments for <b>C</b> ollaborative <b>T</b> ele- <b>S</b> urgery <b>T</b> raining over <b>P</b> hotonic <b>N</b> etworking
ICCE	engl. <i><b>I</b>ntra <b>C</b>apsular <b>C</b>ataract <b>E</b>xtraction</i>

INRIA .....	<b>I</b> nstitut <b>N</b> ational de <b>R</b> echerche en <b>I</b> nformatique et <b>A</b> utomatique
IOP .....	engl. <i>intraocular pressure</i>
LCP .....	engl. <b>L</b> inear <b>C</b> omplementarity <b>P</b> roblem
MKS .....	<b>M</b> ehrkörpersystem
mmHg .....	Einheit für den Druck in Höhe Millimeter-Quecksilbersäule
OLED .....	engl. <i><b>O</b>rganic <b>L</b>ight-<b>E</b>mitting <b>D</b>iode</i>
OMS .....	<b>O</b> perationsmaschinen- <b>S</b> imulation
OpenCL .....	engl. <b>O</b> pen Computing <b>L</b> anguage
pps .....	<b>P</b> ulse pro <b>S</b> ekunde
SI .....	frz. <i><b>S</b>ystème <b>i</b>nternational d'<b>u</b>nités</i>
SOFA .....	<b>S</b> imulation <b>O</b> pen <b>F</b> ramework <b>A</b> rchitecture
SPH .....	engl. <b>S</b> moothed <b>P</b> article <b>H</b> ydrodynamics
ViPA .....	engl. <i><b>V</b>irtual <b>P</b>atient <b>A</b>alysis</i>
VR .....	virtuelle <b>R</b> ealität
VRM .....	engl. <i><b>V</b>irtual <b>R</b>eality in <b>M</b>edicine</i>